

UNIVERSITY of CALIFORNIA  
Santa Barbara

**Estimation and Control with Relative Measurements:  
Algorithms and Scaling Laws**

A Dissertation submitted in partial satisfaction of the  
requirements for the degree

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Prabir Barooah

Committee in charge:

Professor João P. Hespanha, Chair

Professor Bassam Bamieh

Professor Francesco Bullo

Professor Upamanyu Madhow

Professor Prashant G. Mehta

September 2007

The dissertation of Prabir Barooah is approved.

---

Professor Bassam Bamieh

---

Professor Francesco Bullo

---

Professor Upamanyu Madhow

---

Professor Prashant G. Mehta

---

Professor João P. Hespanha, Committee Chair

July 2007

Estimation and Control with Relative Measurements:  
Algorithms and Scaling Laws

Copyright © 2007

by

Prabir Barooah

To my father and my mother

## Acknowledgements

I would like to thank my advisor Professor João P. Hespanha for guiding me through this effort, and for teaching me not only how to conduct research with enthusiasm and rigor, but also how to present results in a clear and concise manner. Professors Bassam Bamieh and Prashant Mehta deserve special mention for the help they provided, without which a large chunk of this dissertation would have been missing. I would also like to thank Professors Upamanyu Madhow and Francesco Bullo for being in my committee.

I am grateful to Professor Petar Kokotović for the excellent training in control theory I received from him. Andrzej Banaszuk supported and helped me in transitioning out of UTRC and Professor Andy Teel helped me in transitioning into UCSB, for that I thank them both. Special thanks go to Kaviyesh, Payam, and Neimar, and especially to Sara, who helped me in ways more numerous than can be listed here.

The material reported in this dissertation is based upon research supported by the Institute for Collaborative Biotechnologies through grant DAAD19-03-D-0004 from the U.S. Army Research Office and by the National Science Foundation under Grant No. CCR-0311084.

# Curriculum Vitæ

Prabir Barooah

## Education

- |           |   |
|-----------|---|
| 1997-1999 | M.S. in Mechanical Engineering, University of Delaware, Newark, DE, USA                 |
| 1992-1996 | B.Tech. in Mechanical Engineering, Indian Institute of Technology, Kanpur, U.P., India. |

## Experience

- |           |   |
|-----------|---|
| 2001–2002 | Senior Engineer/Scientist, United Technologies Research Center, East Hartford, CT, USA.   |
| 1999–2001 | Assistant Research Engineer, United Technologies Research Center, East Hartford, CT, USA. |
| 1996–1997 | G.T. Engineer, Tata Engineering and Locomotive Company, Jamshedpur, India.                |

## Awards

Best Paper Award at the 2nd International Conference on Intelligent Sensing and Information Processing (2005) for the paper titled “Estimation From Relative Measurements : Error Bounds from Electrical Analogy”.

NASA Group Achievement Award (2003) for successful demonstration of active combustion instability control.

## Selected Publications

Prabir Barooah and João P. Hespanha, “Estimation on Graphs from Relative Measurements: Distributed Algorithms and Fundamental Limits”, in *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 57-74, August 2007.

Prabir Barooah and João P. Hespanha. “Estimation from Relative Measurements: Electrical Analogy and Large Graphs”, submitted to *IEEE Transactions on Signal Processing*, April 2007.

Prabir Barooah, Prashant G. Mehta and João P. Hespanha, “Decentralized Control of Vehicular Platoons: Improving Closed Loop Stability by Mistuning”, submitted to *IEEE Transactions on Automatic Control*, Dec. 2006.

Prabir Barooah, Neimar M. Da Silva and João P. Hespanha, “Distributed Optimal Estimation From Relative Measurements for Localization and Time Synchronization”, in Phillip B. Gibbons, Tarek Abdelzaher, James Aspnes, Ramesh Rao, *Distributed Computing in Sensor Systems, vol. 4026 of Lect. Notes in Comput. Science*, June 2006, pp. 266-281.

Prabir Barooah, Prashant G. Mehta and João P. Hespanha, “Control of Large Vehicular Platoons: Improving Closed Loop Stability by Mistuning”, in proceedings of the *American Control Conference*, pp. 4666-4671, July 11-13, 2007.

Prabir Barooah and João P. Hespanha, “Graph Effective Resistance and Distributed Control: Spectral Properties and Applications”, in proceedings of the *45th IEEE Conference on Decision and Control*, December 13-15, 2006, San Diego, pp. 3479-3485.

Prabir Barooah and João P. Hespanha, “Error Amplification and Disturbance

Propagation in Vehicle Strings with Decentralized Linear Control”, in proceedings of the *44th IEEE Conference on Decision and Control*, pp. 4964-4969, Dec. 12-15, 2005.

Prabir Barooah and João P. Hespanha, “Estimation From Relative Measurements : Error Bounds from Electrical Analogy”, in proceedings of the *2nd Int. Conf. on Intelligent Sensing and Information Processing*, pp. 88-93, Jan. 4-7, 2005.

## **Abstract**

Estimation and Control with Relative Measurements:  
Algorithms and Scaling Laws

by

Prabir Barooah

In this dissertation we examine a class of estimation and control problems involving interconnected systems. These problems share the common attribute that, between two component subsystems, noisy measurements of the difference of their states alone is available. The estimation problem is relevant to sensor and actuator networks, and the control problem is relevant to coordination in multi-agent systems. Both classes of problems are defined over a graph that is used to describe the interconnections.

In the first part of this dissertation, the estimation problem is examined. The variables correspond to the nodes of a graph, and the measurements of the noisy difference between pairs of variables correspond to its edges. The task is to compute estimates of the node variables with respect to a reference node. We begin by designing distributed algorithms to compute the optimal estimate, which refers to the best linear unbiased estimator (BLUE). We then examine the effect of the graph structure on the minimum achievable estimation error. Specifically, we examine how the optimal estimation error of a node variable grows with its distance from the reference node. A classification of graphs - sparse and dense in 1D, 2D, and 3D - is obtained, which determines the error growth rate: linear, logarithmic,

or bounded.

In the second part of this dissertation, the control of formations over arbitrary graphs is described. Specifically, we examine how the structure of the interconnection graph affects the stability and sensitivity to measurement noise of the formation. The vehicular platoon problem is investigated in detail - especially the decentralized bidirectional control architecture in which each vehicle uses front and back spacing measurements to compute its control signal. Fundamental limitations in disturbance amplification are established for the symmetric bidirectional architecture. Then we show that arbitrary small asymmetry in the front and back controller gains can lead to an order of magnitude improvement in stability margin.

The underlying theme of our investigations is that of performance degradation – and possible amelioration – in interconnected systems as the the number of constituent sub-systems increases.

# Contents

Acknowledgments	v
Curriculum Vitæ	vi
Abstract	ix
List of Figures	xvi
List of Tables	xx
<b>1 Introduction</b>	<b>1</b>
1.1 Part I : Estimation with relative measurements . . . . .	4
1.1.1 Challenges and contributions . . . . .	6
1.2 Part II : Decentralized control with relative measurements . . . .	15
1.2.1 Contributions . . . . .	16
1.3 Notation . . . . .	21
<b>2 Estimation with relative measurements: applications and the optimal estimator</b>	<b>24</b>
2.1 Applications . . . . .	25
2.1.1 Sensor network localization . . . . .	25
2.1.2 Time synchronization . . . . .	29
2.1.3 Motion Consensus . . . . .	33
2.1.4 Mobile robot localization . . . . .	34
2.2 Measurement graph and optimal estimation . . . . .	36

2.2.1	The optimal estimator (BLUE) and the optimal estimates	37
2.2.2	Dirichlet Laplacian and BLUE . . . . .	45
2.2.3	Role of parallel edges . . . . .	48
2.3	Proofs . . . . .	49
<b>3</b>	<b>Distributed algorithms for optimal estimation</b>	<b>54</b>
3.1	Problem statement . . . . .	55
3.2	Contribution and prior work . . . . .	60
3.3	Jacobi algorithm . . . . .	64
3.3.0.1	Asynchronous implementation . . . . .	69
3.3.1	Correctness and performance analysis of the Jacobi algorithm	72
3.3.1.1	Correctness of the Jacobi algorithm . . . . .	77
3.3.1.2	Convergence rate of the Jacobi algorithm . . . . .	84
3.3.2	Reducing error faster - flagged initialization . . . . .	85
3.3.3	Simulations of the Jacobi algorithm . . . . .	86
3.4	The overlapping subgraph estimator(OSE) algorithm . . . . .	90
3.4.0.1	Modified EPA . . . . .	95
3.4.1	Simulations of the OSE algorithm . . . . .	96
3.4.2	Energy cost comparison . . . . .	98
3.5	Effect of asymmetric communication . . . . .	102
3.5.1	An impossibility result . . . . .	102
3.5.2	More measurements need not reduce error . . . . .	103
3.6	Comments and open problems . . . . .	104
3.7	Proofs . . . . .	108
<b>4</b>	<b>Optimal estimation in infinite graphs and electrical analogy</b>	<b>114</b>
4.1	Introduction . . . . .	114
4.2	Problem statement . . . . .	116
4.3	Contributions and prior work . . . . .	119
4.4	Generalized electrical networks . . . . .	121
4.4.1	Existence and uniqueness of generalized current . . . . .	122
4.4.2	Generalized effective resistance . . . . .	124

4.5	Main results . . . . .	125
4.5.1	Convergence rate . . . . .	130
4.6	Properties of generalized electrical networks . . . . .	133
4.6.1	Rayleigh's monotonicity law . . . . .	133
4.6.2	Other properties . . . . .	136
4.6.3	Approximating infinite network currents . . . . .	137
4.6.4	Electrical analogy for finite networks . . . . .	138
4.7	Proof of the main results . . . . .	140
4.8	Comments and open problems . . . . .	143
4.9	Technical proofs . . . . .	144
<b>5</b>	<b>Error scaling laws</b>	<b>158</b>
5.1	Introduction . . . . .	158
5.2	Contributions and prior work. . . . .	161
5.3	Problem statement and main results . . . . .	164
5.3.1	Graph denseness and sparseness . . . . .	165
5.3.1.1	Graph drawing . . . . .	165
5.3.1.2	Dense and Sparse Graphs . . . . .	168
5.3.2	Error scaling laws . . . . .	170
5.3.3	Counterexamples to conventional wisdom . . . . .	171
5.4	Dense and sparse graphs . . . . .	172
5.4.1	Relationship with lattices . . . . .	174
5.4.2	Checking denseness and sparseness . . . . .	176
5.5	Establishing the error scaling laws . . . . .	178
5.5.1	Effective resistance for lattices and fuzzes . . . . .	179
5.5.2	An intuitive explanation . . . . .	180
5.5.3	Proof of the Error Scaling Laws Theorem 5.3.1 . . . . .	183
5.6	Comments and open problems . . . . .	186
5.7	Proofs . . . . .	188
<b>6</b>	<b>Decentralized formation control: effective resistance vs. scalability</b>	<b>202</b>

6.1	Introduction . . . . .	202
6.2	Contributions and prior work . . . . .	205
6.3	Formation control with noisy measurements . . . . .	208
6.3.1	Implications for man-made autonomous agents . . . . .	213
6.3.2	Implications for Swarming in Nature . . . . .	214
6.4	Lower bound on the Dirichlet Laplacian spectrum from effective resistance . . . . .	217
6.4.1	Role of the Dirichlet Laplacian spectrum . . . . .	217
6.4.1.1	Convergence rate of discrete-time algorithms . . . . .	217
6.4.1.2	Convergence rate of continuous-time algorithms . . . . .	218
6.4.2	Effective resistance between a node and a set of nodes . . . . .	219
6.4.3	Lower bound from effective resistance . . . . .	220
6.5	Comments and open problems . . . . .	222
<b>7</b>	<b>Control of vehicular platoons: symmetric bidirectional control</b>	<b>224</b>
7.1	Introduction . . . . .	224
7.2	Prior work and contributions . . . . .	227
7.2.1	Prior work on vehicular platoons . . . . .	227
7.2.2	Main results . . . . .	230
7.3	Problem statement and main results . . . . .	232
7.3.1	Closed loop stability with symmetric bidirectional control . . . . .	235
7.3.2	Steady-state errors . . . . .	236
7.3.3	Effect of lead vehicle's deviation from constant velocity . . . . .	237
7.3.4	Disturbance propagation . . . . .	239
7.3.5	Explanation through graph eigenvalues . . . . .	240
7.4	Comments and open problems . . . . .	242
7.5	Proofs . . . . .	244
<b>8</b>	<b>Control of vehicular platoons: asymmetric bidirectional control</b>	<b>253</b>
8.1	Introduction . . . . .	253
8.2	Contributions and prior work . . . . .	254
8.3	Problem statement . . . . .	256

8.4	Continuous model of an automated platoon . . . . .	261
8.4.1	Eigenvalue comparison . . . . .	265
8.5	Loss of stability margin with symmetric bidirectional control . . .	266
8.6	Reducing loss of stability by mistuning . . . . .	273
8.6.1	Perturbation analysis . . . . .	274
8.6.2	Eigenvalue optimization . . . . .	283
8.7	Simulations . . . . .	288
8.8	Comments and open problems . . . . .	291
8.9	Proofs . . . . .	292
<b>9</b>	<b>Summary</b>	<b>296</b>

# List of Figures

1.1	A measurement graph and two separate paths from the reference to node 4. . . . .	7
1.2	Two measurement graphs with very different error scaling laws. . .	12
2.1	Relative position measurement in a Cartesian reference frame using range and bearing measurements. . . . .	27
2.2	Relative position measurement between two cameras in a Cartesian reference frame by looking at a common object. . . . .	28
2.3	Measurement of differences in local times by bidirectional exchange of time-stamped messages. . . . .	31
2.4	RBS: measurement of differences in local times by unidirectional message exchange [1]. . . . .	32
2.5	A group of mobile robots whose current positions are to be estimated from relative position measurements between robots at the same time and between the same robots at successive times. . . .	35
2.6	A measurement graph $\mathcal{G}$ and a few of its associated matrices. . .	40
2.7	Dirichlet Laplacian for the graph in Figure 2.6. . . . .	41
2.8	An example of a measurement graph, the BLUE estimates and error covariances. . . . .	42
2.9	Figure 2.8 contd. . . . .	43
3.1	A measurement graph and a symmetric communication graph associated with it. . . . .	59
3.2	A measurement graph and an asymmetric communication graph associated with it. . . . .	59
3.3	The Jacobi Iteration with symmetric communication. . . . .	67

3.4	Figure 3.3 contd. . . . .	68
3.5	The Jacobi Iteration with asymmetric communication. . . . .	70
3.6	The matrices used to compactly represent the Jacobi algorithm with asymmetric communication. . . . .	75
3.7	Figure 3.6 contd. . . . .	76
3.8	A measurement graph created by an ad-hoc sensor network with 200 nodes distributed randomly in a unit square area. . . . .	88
3.9	Simulation results of the Jacobi algorithm with symmetric communication. . . . .	89
3.10	Simulation results on the convergence of the Jacobi algorithm with asymmetric communication. . . . .	90
3.11	A measurement graph and a 2-hop subgraph centered at node 4 for explaining the OSE algorithm. . . . .	93
3.12	Performance comparison between the Jacobi and OSE algorithms, and convergence of OSE with communications faults. . . . .	97
3.13	Comparison between the three algorithms in terms of average energy consumption. . . . .	101
3.14	Example showing the impossibility of any distributed algorithm to converge to the optimal estimate when communication is asymmetric. . . . .	103
3.15	An example of more measurements reducing the estimation accuracy of the Jacobi algorithm when communication is asymmetric. . . . .	105
4.1	A nested sequence of measurement graphs that “tend to” the 2-dimensional square lattice. . . . .	127
4.2	A sequence of finite, nested subgraphs of the 2D lattice, and the sequence of BLUE variances of a particular node variable. . . . .	131
5.1	A drawing of a graph in 2D Euclidean space, and the corresponding denseness and sparseness parameters . . . . .	168
5.2	Three measurement graphs that show vastly different scaling laws of the estimation error, whereas each has the same node degree for every node. Furthermore, they are all “sparse” according to traditional graph-theoretic terminology (see the discussion on graph denseness in Section 5.1). . . . .	173
5.3	Lattices. . . . .	174

5.4	Doyle and Snell [2]’s illustration on approximating lattices by Euclidean spaces. . . . .	181
5.5	Superimposing a 2-dimensional lattice (brown) on a 2-dimensional dense graph (black). . . . .	190
5.6	Natural drawing of the 2-D lattice (brown) superimposed on the $f'$ drawing of $\mathcal{G}$ . Edges are not shown to prevent clutter. In this example, $u = \bar{u}$ but $v \neq \bar{v}$ . . . . .	194
5.7	A graph that can be drawn in a civilized manner in $\mathbb{R}^2$ can be embedded in a fuzz of the 2-D lattice. . . . .	196
5.8	Fuzzing doesn’t change the effective resistance too much. . . . .	200
6.1	A symmetric interconnection graph, its generalized incidence matrix $\mathcal{A}$ , and the matrix $\mathcal{B}$ . . . . .	211
6.2	Examples of 1-D and 3-D network topologies in natural swarms . . . . .	215
6.3	The relationship between algebraic connectivity and the smallest eigenvalue of the Dirichlet Laplacian. . . . .	221
7.1	platoon of vehicles . . . . .	225
7.2	The interconnection architecture graph for symmetric bidirectional control. . . . .	232
7.3	Nyquist plot of $H(s)K(s)$ when $H(s)K(s)$ has two integrators. . . . .	241
7.4	Nyquist plot of $H(s)K(s)$ when $H(s)K(s)$ has one integrator. . . . .	242
8.1	A platoon with $N$ vehicles moving in one dimension. . . . .	256
8.2	Comparison of closed loop eigenvalues of the platoon dynamics and the spectrum of the corresponding PDE. . . . .	267
8.3	A schematic explaining the loss of stability as $N$ increases and how mistuning ameliorates this loss. . . . .	269
8.4	Comparison of the least stable eigenvalue of the closed loop platoon dynamics and that predicted by the PDE with symmetric bidirectional control. . . . .	272
8.5	Benefit of mistuning design on the closed loop spectra in the fictitious-leader-and-follower case. . . . .	277
8.6	Benefit of mistuning design on the closed loop spectra in the fictitious-leader-only case. . . . .	279
8.7	Comparison of six eigenvalues of the platoon and the PDE. . . . .	282

8.8	The optimal mistuning pattern $k_m^*(x)$ for the symmetric PDE. . .	287
8.9	Least stable eigenvalue with the optimum mistuning profile computed for the symmetric PDE. . . . .	287
8.10	Simulations with symmetric bidirectional control. . . . .	289
8.11	Simulations with mistuned bidirectional control. . . . .	290

# List of Tables

1.1	Notation . . . . .	22
5.1	Answer to the error scaling question provided by Theorem 5.3.1. .	170
5.2	An intuitive explanation of error scaling laws by continuum approximation . . . . .	183
8.1	The two scenarios – one with a fictitious lead vehicle, the other with fictitious lead and follow vehicles. . . . .	258
8.2	The eigen-solutions for the Laplacian with two different boundary conditions. . . . .	268
8.3	the trend of the less stable eigenvalue $s_l^+$ for the PDE (8.18) . . .	270

# Chapter 1

## Introduction

Recent years have seen the development and proliferation of devices that are equipped with embedded sensing, processing, wireless communication, and actuation capability. As a result, it is becoming possible to monitor and control processes that are distributed over large geographical areas, by deploying a large number of such devices and interconnecting them, possibly using wireless communication. The individual devices are called *nodes*, and the collection of such devices deployed for sensing tasks is called a *sensor network*. The epithet network refers to the interconnection between nodes. The New York Harbor Observing and Prediction System (NYHOPS) is an example of a sensor network, which consists of a number of sensors connected by a wireless communication network that monitor salinity, temperature, turbidity, and water levels in the Hudson river estuary [3]. Another example of a sensor network is a group of unmanned aerial vehicles (UAVs) that can be used to collaboratively detect and track targets [4]. When the devices have actuation capability as well, the network is called an *actuator network*. More generally, a network of devices with both sensing and actuation capability is called a *sensor-actuator network*. An example of sensor-

actuator networks is an irrigation network of water level sensors and gate actuators that is interconnected over a communication infrastructure [5]. Another example of a sensor-actuator network is an Automated Highway System (AHS), in which a group of autonomous vehicles forms a platoon, in which every vehicle takes local control action based on on-board sensor readings, so that a constant inter-vehicular separation is maintained [6, 7].

Successful application of sensor and actuator networks requires tackling novel estimation and control problems. Typically, the nodes of a sensor and actuator network are distributed in a large spatial domain. As a result, only local information is available to each of the nodes. The nodes have to either estimate global quantities of interest, or take appropriate control action, based on locally measured quantities. A special situation that arises in several applications is that a node can measure *relative* quantities, from which it has to estimate the absolute ones. For example, a node maybe able to measure its relative position with respect to a nearby node but not its position in a global coordinate. The small size, low cost, and low energy budget of the sensor nodes that make them so attractive for a myriad of applications preclude these nodes having on-board GPS [8]. Yet, in order that the user of the network can utilize the data gathered by the sensor network, location information of the data sources is needed. Therefore, nodes need to estimate their own locations in a global coordinate frame from measurements of relative positions, which may furthermore be corrupted with high levels of noise due to the limitations of the measurement techniques.

In certain applications, estimation of global attributes may not be needed, but the nodes may need to take appropriate control action based solely on relative measurements in order to achieve a common objective. For example, a team of UAVs may need to maintain a specific formation, while each UAV can measure

only its relative position with respect to its nearby UAVs. Each node, i.e., UAV, is required to employ a decentralized control law that uses only the local, relative position measurements. Even if the dynamics of individual nodes were otherwise independent of each other, since the control action taken by one node depends on its relative position with other nodes, the closed-loop dynamics of the nodes become coupled with one another.

Fusing noisy measurements obtained from a network of nodes to produce accurate estimates, when the nodes are spatially separated, communicate with one another through an unreliable wireless medium, and have limited battery lives, is a challenging task – especially when the size of the network is large. Similarly, devising local control algorithms for individual nodes of an interconnected network of dynamical systems, such that the whole system achieves a global objective, is a difficult problem, one for which traditional design and analysis tools of control theory are not adequate.

This dissertation investigates a few of the estimation and control problems that are motivated by sensor and actuator network applications. The dissertation consists of two parts – part I deals with the estimation problems and part II, with control. In the following sections we describe the problems examined in the two parts, the challenges in each, and briefly summarize the contributions of this work in each problem category. The contributions are listed chapter-wise for easy referral. Each chapter contains, at its end, a discussion of open issues.

## 1.1 Part I : Estimation with relative measurements

We examine the problem of estimating  $n$  variables  $x_1, x_2, \dots, x_n$  from noisy relative measurements of the form

$$\zeta_{u,v} = x_u - x_v + \epsilon_{u,v}, \quad u, v \in \{1, \dots, n\}, \quad (1.1)$$

where  $\epsilon_{u,v}$  is zero-mean measurement-noise, when one or more variables is assumed known. This problem arises in several sensor network applications. The variables are often vector-valued.

A typical example is localization, in which locations of a number of nodes have to be estimated from relative position measurements of the form (1.1). The need for localization arises when the nodes cannot measure their positions directly, such as when they are not equipped with global positioning service (GPS) capability. However, certain pairs of nodes may be able to measure their relative positions. For example, two nodes  $u$  and  $v$  that are located at positions  $x_u$  and  $x_v$  on a plane may be able to measure their relative position  $x_u - x_v$  in a common Cartesian coordinate frame. The details of acquiring such measurements in practice, using either on-board wireless devices, or vision based sensors, are described in Section 2.1.1. Irrespective of the technique used, the relative position measurement so obtained will have errors, and hence can be expressed in the form of (1.1).

Apart from localization, there are several problems relevant to sensor and actuator network applications where variables are to be estimated from noisy relative measurements of the form (1.1), which include time-synchronization and motion consensus. In time-synchronization, a network of nodes whose local clocks progress at varying speeds (skews) and have different offsets from one another

need to be synchronized. Two nodes that can communicate with each other can obtain noisy measurements of the difference between their offsets and ratio of their skews. The skews and offsets of all the nodes with respect to a common reference need to be estimated from the noisy relative measurements.

In motion consensus, a group of mobile nodes need to estimate their velocities or directions with respect to a leader, but each node can only measure its relative velocity with respect to a few nearby neighbors. In Chapter 2 we will describe these problems in detail.

Each variable  $x_u$  to be estimated is called a *node variable*, which is associated with the node  $u$ . The known node variables are called *reference variables*. The  $\zeta_{u,v}$  in (1.1) is called a *relative measurement*, or sometimes simply *measurement*. Node variables in general are vector valued, and the dimension of a node variable is denoted by  $k$ . For example, in the localization problem  $k$  can be 2 or 3, depending on whether the nodes are located in a 2D or 3D space. Note that although in general nodes are located in 3D space, sometimes the third dimension may be irrelevant. This estimation problem can be naturally associated with a *measurement graph*  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ . The vertex set  $\mathcal{V}$  of the measurement graph consists of the set of nodes  $\mathcal{V} := \{1, \dots, n\}$ , where  $n$  is the number of nodes, while its edge set  $\mathcal{E}$  consists of all of the ordered pairs of nodes  $(u, v)$  such that a noisy measurement of the form (1.1) between  $u$  and  $v$  is available. The measurement errors on distinct edges are assumed uncorrelated. The variables that are known are called *reference variables*. In practice, none of the variables may be known, in which case we arbitrarily assign one of the nodes, say  $o \in \mathcal{V}$ , to be the reference node, and set  $x_o = 0$ . The measurement graph  $\mathcal{G}$  is a directed graph since  $(u, v) \in \mathcal{E}$  implies the measurement  $\zeta_{u,v}(= x_u - x_v + \epsilon_{u,v})$  is available while  $(v, u) \in \mathcal{E}$  implies the measurement  $\zeta_{v,u}(= x_v - x_u + \epsilon_{v,u})$  is, and these two

measurements are distinct.

### 1.1.1 Challenges and contributions

The error  $\epsilon_e$  affecting the measurement  $\zeta_e$  (where  $e \in \mathcal{E}$ ) can be quite large depending on the application and sensing technology. The goal is to obtain accurate estimates of all the node variables from the noisy measurements. An estimate of a node variable  $x_u$  can be obtained by adding the measurements (after appropriately modifying their signs) along a path from  $u$  to a reference node. For example, in Figure 1.1, consider the undirected path  $\mathcal{P}_1 := \{1, e_2, 2, e_4, 3, e_5, 4\}$  from the reference node 1 to the node 4. By adding the measurements along the path, we obtain an estimate of  $x_4$ :

$$\begin{aligned}\hat{x}_4 &= -\zeta_2 - \zeta_4 + \zeta_5 = -(x_2 - x_1 + \epsilon_2) - (x_2 - x_3 + \epsilon_4) + (x_4 - x_3 + \epsilon_5) \\ &= x_4 + (-\epsilon_2 - \epsilon_4 + \epsilon_5)\end{aligned}$$

where  $x_1$  vanishes since the reference variable  $x_1$  is assumed to be 0. Since the measurements errors are assumed uncorrelated, the covariance of the estimation error is the sum of the covariances of the measurement errors  $\epsilon_2, \epsilon_4$ , and  $\epsilon_5$ . Denoting by  $P_e$  the covariance of the measurement error  $\epsilon_e$ , i.e.,  $P_e := \mathbb{E}[\epsilon_e \epsilon_e^T]$ , the covariance of the error in the estimate  $\hat{x}_4$  is

$$\mathbb{E}[(\hat{x}_4 - x_4)(\hat{x}_4 - x_4)^T] = P_2 + P_4 + P_5.$$

However, it is possible to construct another estimate of  $x_4$  by using measurements along another path,  $\mathcal{P}_2 := \{1, e_1, 2, e_3, 4\}$ , from the reference 1 to 4:

$$\hat{\hat{x}}_4 = -\zeta_1 - \zeta_3 = -(x_1 - x_2 - \epsilon_1) - (x_2 - x_4 + \epsilon_3) = x_4 + \epsilon_1 - \epsilon_3.$$

The covariance of the error in the estimate  $\hat{\hat{x}}_4$  is  $\mathbb{E}[(\hat{\hat{x}}_4 - x_4)(\hat{\hat{x}}_4 - x_4)^T] = P_1 + P_3$ .

If all the measurement error covariances are equal, the error covariance in the

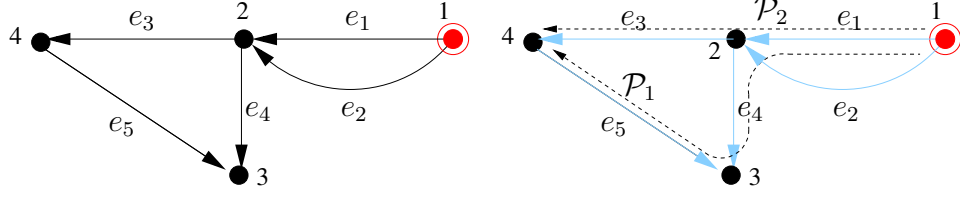


Figure 1.1. Estimating node variables by adding relative measurements along a path. Two different paths,  $\mathcal{P}_1$  and  $\mathcal{P}_2$ , are shown, that go from 1 to 4.

estimate  $\hat{x}_4$  is smaller than the error covariance in the estimate  $\hat{x}_4$ . Still, both the estimates above uses only a subset of the available measurements. It is possible to construct a more accurate estimate by using all the available measurements.

We will describe in Chapter 2 how the *optimal estimate* of all the node variables can be computed by using all the measurements. The optimal estimate refers to that obtained by the classical best linear unbiased estimator (BLUE) that is guaranteed to produce the minimum variance estimate among all linear unbiased estimators [9]. When all the measurements, their associated error covariances, and information about the measurement graph  $\mathcal{G}$  are available at a single processor, it can compute the optimal estimates. Therefore the estimation problem described above can be solved by first sending all measurements to one particular node, computing the optimal estimates in that node, and then distributing the estimates to the individual nodes.

However, this centralized solution is undesirable for several reasons. First, when wireless communication is used, this method unduly burdens the nodes close to the central processor. In a large ad-hoc network of wireless nodes, sending all of the measurements requires multi-hop communication, and most of the data transmitted to the central processor have to be routed through the nodes close to it. When the nodes operate on batteries with small energy budgets, this

mode of operation greatly reduces the life of the nodes that carry out most of the communication. It should be noted that the primary source of energy consumption in wireless sensor networks is communication [10], while much less energy is consumed for computation [11]. Second, centralized computation is less robust to node and link failures over time. Multi-hop data transfer to a central node typically requires the construction of a routing tree rooted at the central node. Failure of a node in one of the branches of the routing tree effectively cuts off communication from all of the nodes in the tree branch rooted at the faulty node. In addition, construction of a routing tree can be challenging when communication links suffer from temporary failures or when nodes are mobile [12]. Third, a centralized computation renders the entire network susceptible to a catastrophe if the central processor fails. This discussion raises a key issue that is investigated in this dissertation:

***Question 1 (distributed estimation):*** *Is it possible to construct the optimal estimate in a distributed fashion such that the communication and computation burden is shared equally by all the nodes? If so, how much communication is required between nodes, and how robust is the distributed algorithm with respect to communication failures?*

By a *distributed algorithm* we mean an algorithm in which every node carries out independent computations to estimate its own variable, but is allowed to periodically exchange messages with a set of neighbors that are close enough to it so as to enable communication. We show that it is indeed possible to design distributed algorithms to compute optimal estimates that are robust to communication faults. Our contributions in this regard are briefly outlined below:

1. In Chapter 3 we develop and analyze two distributed algorithms - Jacobi and OSE - for computing the optimal estimates when the measurement graph

does not change with time. These algorithms are iterative, and the estimates produced by the algorithms converge to the optimal ones if inter-node communication is symmetric, i.e., if a node  $u$  can receive messages from another node  $v$ , then  $v$  can also receive messages from  $u$ . The convergence of the algorithms are proved to be robust to the presence of temporary communication failures. We relate the convergence rate of the Jacobi algorithm to the spectral properties of the measurement graph, in particular, to the minimum eigenvalue of a matrix that describes the structure of the measurement graph.

2. In Chapter 3, we also examine the effect of asymmetric communication, which refers to the situation where a node  $u$  can receive messages from another node  $v$ , but  $v$  cannot receive messages from  $u$ . Such asymmetry is especially common in ad-hoc wireless networks on account of inhomogeneous interference, packet collisions, and inaccurate time synchronization. We show that in presence of asymmetric communication, the Jacobi algorithm still converges, but to a sub-optimal estimate.

The convergence rate of the algorithms proposed here can be slow in large graphs. In addition, the distributed algorithms we have proposed are not applicable when the measurement graph changes with time, e.g., when positions of mobile nodes are to be estimated and node variables are dynamically evolving in time. Therefore, in certain situations, a centralized computation, or perhaps a combination of distributed and centralized computation, may still be required.

Irrespective of which estimation algorithm is used to estimate the node variables, no linear unbiased estimator can obtain an estimate that is more accurate than that of the BLUE. This offers a compelling reason to study the error in

the BLUE, since the accuracy of the BLUE provides a fundamental limit to the accuracy achievable by any estimation algorithm.

Sensor networks consisting of more than a thousand nodes have already been developed [13]. Furthermore, it is envisioned that sensor networks consisting of tens of thousands, if not millions, of nodes are going to be deployed in the near future [10]. Therefore, understanding limits of performance in large graphs, particularly limits that are algorithm-independent, is important if such networks are to be deployed successfully.

For these reasons, we study large graphs with a single reference node and examine how the error covariance of a node variable's estimate changes as the node's distance from the reference increases. In general, one expects the error to grow with distance. The growth rate determines the size of the graph that can be "serviced" by a single reference node. For a given acceptable estimation error, if the error growth rate is high, either the graph has to be kept small, or more reference nodes have to be introduced.

We show that the structure of the measurement graph determines, to a great extent, how the estimation error of a node will vary as a function of its distance from the reference node. Evidence in support of this statement is presented in Figure 1.2, which shows two graphs and the optimal estimation error variances of the node variables in each. Both the graphs  $\mathcal{G}_A$  and  $\mathcal{G}_B$  are obtained by placing nodes randomly in the plane and allowing two nodes to have an edge between them if and only if their distance is less than a certain value. In case of  $\mathcal{G}_A$ , nodes are allowed to fall only within the boundary shown in dashed lines. One can think of  $\mathcal{G}_A$  as a sensor network obtained by placing nodes randomly in an urban terrain, whereas  $\mathcal{G}_B$  is obtained by placing nodes in a large, level area. Both graphs have one reference node, placed at  $(0,0)$ , the same average node degree

of 3.2, the same node density of 500 nodes per unit area of the deployed region, and the same measurement error variance on each edge. The degree of a node is the number of edges incident on it, therefore same node degree implies the same number of measurements per variable. For simplicity, we considered the case of scalar node variables. Computation of the optimal estimation error variances is described in Chapter 2. From the plot of the estimation error variances, we see that the graphs have quite distinct estimation error growth rates.

This example motivates the second issue investigated in this dissertation:

***Question 2 (error scaling):*** *Can different graphs exhibit vastly different error scaling laws, i.e., the rate at which the optimal estimation error covariance of a node variable grows as a function of the node's distance from the reference? If so, what structural properties of measurement graphs determine these scaling laws, and how do we identify these properties as well as the scaling laws?*

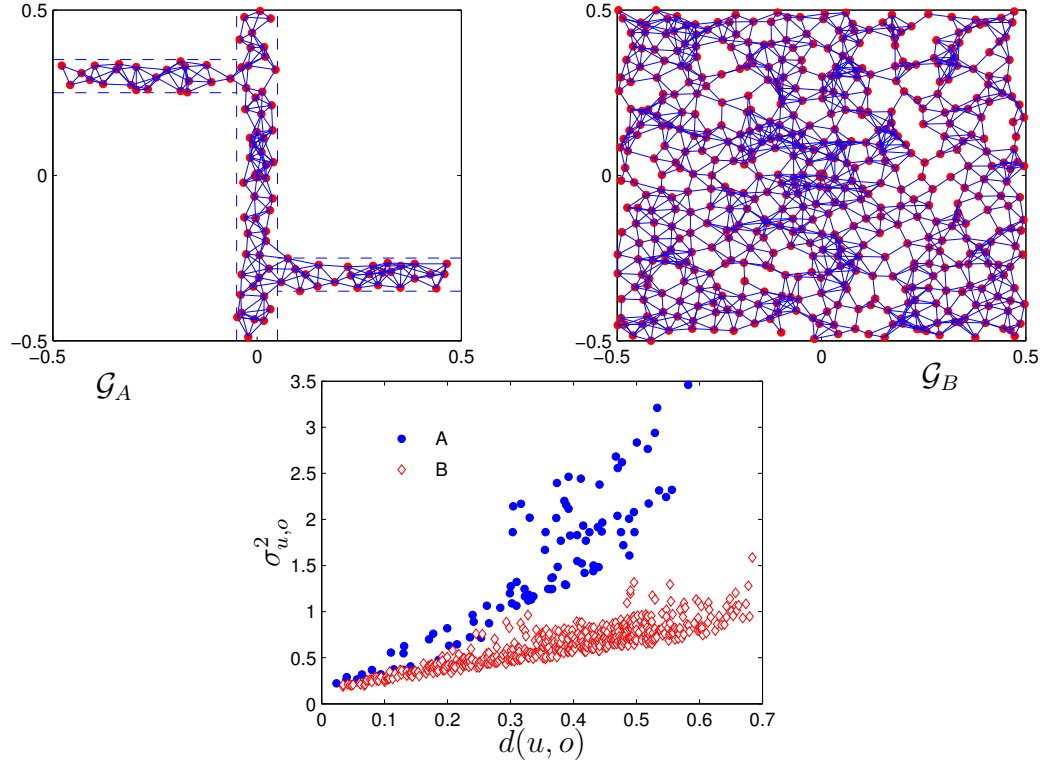


Figure 1.2. Two measurement graphs with very different error scaling laws. The two graphs,  $\mathcal{G}_A$  and  $\mathcal{G}_B$  are obtained by placing nodes randomly in the plane and allowing two nodes to have an edge between them if their distance is less than a certain value. In case of  $\mathcal{G}_A$ , nodes are allowed to fall only within the boundary shown as dashed lines. Both graphs have one reference node, placed at  $(0, 0)$ . Both graphs have the same average node degree, namely 3.2, the same node density, namely 500 nodes/unit area, and the same measurement error variance, namely 1, on each edge. The bottom plot shows the trend of the optimal estimation error variances in the two graphs as a function of the Euclidean distance  $d(u, o)$  between the nodes  $u$  and  $o$  in the plane. The legend  $A$  refers to the graph  $\mathcal{G}_A$ , and  $B$  refers to the graph  $\mathcal{G}_B$ .

The question of error scaling is important to study for a number of reasons. Given a maximal acceptable error, the number of nodes whose estimation errors are lower than this level is large if the graph exhibits a slow increase of variance with distance, but small otherwise. These scaling laws therefore help one design and deploy large networks for which accurate estimation is possible. In addition, knowledge of the scaling laws for the optimal estimation error provide an algorithm-independent limit on the lowest possible error growth, since the optimal estimator has the lowest estimation error variance among all linear estimators.

We show that estimation error can indeed exhibit vastly different scaling laws depending on certain structural properties of the graph. Our contributions in this regard are briefly outlined below:

1. In Chapter 4, we examine infinite measurement graphs, in which the number of variables and measurements are countably infinite. We show that under certain conditions, the estimation error covariance of a node variable in a large finite graph is close to that in an infinite graph. Intuitively, for a node that resides sufficiently inside a large finite graph, i.e., not close to the boundary, the graph appears to extend to infinity in all directions. The results in Chapter 4 provides formal justification for using infinite graphs as proxies for large finite graphs and also establishes the conditions under which such an approximation is valid. The advantage of working with infinite graphs is that boundary conditions in infinite graphs are weaker than in finite graphs, which make them easier to analyze.
2. As a first step toward answering the error-scaling question, we prove in Chapter 4 that the covariance of a node's optimal estimation error is equal to the *matrix-valued effective resistance* in an abstract electrical network

that can be constructed from the measurement graph and measurement error covariances.

3. In Chapter 5 We obtain a classification of graphs, namely, dense or sparse in  $\mathbb{R}^d$ ,  $1 \leq d \leq 3$ , that determines the error scaling laws. In particular, if a graph is dense in 1,2, and 3D, then a node variable's estimation error is upper bounded by a linear, logarithmic, and bounded function of distance from the reference. Corresponding lower bounds are obtained if the graph is sparse in 1, 2 and 3D. The electrical analogy is instrumental in obtaining the error scaling laws. The sparse graphs are simply the “graphs that can be drawn in a civilized manner” that were originally introduced by Doyle and Snell [2].

That the error grows with distance without bound in many graphs is perhaps not surprising. What is perhaps surprising is that there are graphs in which the covariance remains below a constant value regardless of the distance.

Analogies with electrical networks are used in [2, 14] to construct elegant solutions to various graph problems, notably those concerned with random walks. In [2], questions about random walks in certain infinite graphs are answered by bounding the effective resistance in those graphs with that in lattices. It turns out that a similar approach can be used to answer the question of error scaling, once we establish the analogy between error covariance matrices and matrix-valued effective resistances.

We note that scaling laws of the estimation error but are not captured by naive measures of density such as node degree or node and edge density, which are commonly used in the sensor networks literature [15–17]. Using the dense and sparse classification obtained in this dissertation, we provide counterexamples that

expose certain misconceptions that exist in the sensor network literature about the relationship between graph structure and estimation error. These counterexamples show that graphs with the same node degree can exhibit vastly different error scaling laws.

## 1.2 Part II : Decentralized control with relative measurements

In a number of applications, teams of mobile autonomous agents are required to perform tasks in a collaborative manner. For example, consider a team of autonomous mobile nodes (UAVs, ground robots, underwater vehicles etc.) that are required to maintain a particular formation while in motion. The formation is specified in terms of desired relative positions between every pair of nodes. Each node is allowed to communicate and measure its relative position with only a small subset of all the nodes. Each node's task is to take control actions using locally available measurements, such that the group still attains its collective goal of maintaining the desired formation.

Motivation for studying formation control problems arises from their relevance in a wide spectrum of problems, from military surveillance to swarming in nature. Maintaining a formation while in motion can reduce aerodynamic drag in aircrafts [18, 19] and allegedly in birds and spiny lobsters as well [20–22], increase traffic capacity in highways [6], ensure full coverage of the sensed field in spite of limited sensing capability of individual nodes [23], and build extra-terrestrial interferometric imaging system composed of multiple satellites [24].

In all these situations, whether man-made or natural, it is reasonable to as-

sume that the individual nodes have access to only relative position or velocity measurements. The problem of formation control using only locally available measurements falls under the broader category of *decentralized coordination* problems [25]. In such problems a group of nodes have to achieve a common objective without the help of a central authority, while nodes have access to limited, local information. In this dissertation we use the term *decentralized control architecture* to refer to the architectural constraint that each node in an interconnected system is allowed to use, as input to its local control algorithm, information that is available to it or that it collects by communicating with a few nearby neighbors. This is in contrast to a centralized architecture in which information gathered by every node is made available to a central controller that computes appropriate control actions for all the nodes. In an interconnected system of many constituent nodes that are spatially separated, a decentralized architecture is desirable over a centralized one since the latter suffers from large communication overhead. In large interconnected systems in particular, such overhead may make a centralized architecture well-nigh impossible, making decentralized architectures the only possibility.

### 1.2.1 Contributions

In problems of decentralized coordination, including the specific problem of formation control, the task of the design engineer is to develop control algorithms for all the nodes of a sensor-actuator network, so that every node implements a local control law that uses only locally available information, while the network still attains its collective goal. In a general interconnected system, this leads to a high degree of complexity. For example, if there are  $n$  agents that make up the interconnected system, in principle  $n$  separate control algorithms can be designed,

one for each agent. However, tools for designing controllers for an interconnected system which incorporate the constraint of decentralized architecture explicitly are not well-developed (with a few notable exceptions such as [26–28]). As a result, the designer is often forced to resort to a somewhat ad-hoc design, based on trial and error. Frequently, it is arbitrarily decided that every agent will use the same control algorithm, so that the problem is reduced to that of designing a single control algorithm. In fact, a large number of control laws examined in the literature on sensor-actuator network falls into this category [29, 30]. Even with such simplification, performance analysis of the closed loop system is not straightforward due to lack of appropriate analysis tools. In summary, a clear understanding of the effect of interconnection structure on performance, such as stability and robustness to measurement noise etc., is lacking.

Motivated by these issues, we have examined the problem of decentralized formation control from relative measurements. We outline below the specific aspects examined and our contributions:

1. In Chapter 6, we study the formation control problem with relative measurements. We quantify the effect of interconnection topology on noise-sensitivity of the closed loop formation. To focus on the effect of interconnection, simple forms of node dynamics and control laws are assumed. We examine a control law that uses only relative measurements, which has been extensively used in the literature.

For this formation control problem, we show that the covariance of the steady state error (on account of measurement noises) is equal to a matrix-valued effective resistance in an abstract electrical network that can be constructed from the formation graph. This effective resistance was introduced

earlier in chapter 4, and the effect of graph structure on effective resistance was studied in Chapter 5. Using the results from those chapters, we show that the performance of the algorithm in presence of noise is quite sensitive to interconnection topology, and show in which graphs formation control with small errors is possible and in which graphs it is not possible. The analogy with effective resistance is used to explain certain observations in animal formations.

2. The stability margin of the formation is shown to depend on the least stable eigenvalue of the Dirichlet Laplacian matrix of the interconnection graph. This matrix, originally encountered in the estimation problem studied in part I, naturally arises in control and estimation problems of relevance to interconnected systems. The minimum eigenvalue of the Dirichlet Laplacian matrix also determines the rate of convergence of many algorithms used in control and estimation problem (in both continuous and discrete-time settings). This eigenvalue essentially captures the rate of information propagation through the graph, which makes it a key player in the convergence rate analysis of such algorithms.

In Chapter 6 we obtain a bound on this eigenvalue in terms of the matrix-valued effective resistance introduced earlier in Chapter 4. It turns out that effective resistances provide a non-trivial lower bound on this eigenvalue, for which few tools are otherwise available.

3. In Chapter 7, we examine the the problem of decentralized control of vehicular platoons, in which the control objective is to maintain a constant inter-vehicular separation. For this problem, we allow more complex controllers and dynamics, but study a specific interconnection topology. Interest in the

control of platoons has a long history, dating back at least half a century (see the 1958 paper [31]).

The particular architecture we study in this chapter is known as symmetric bidirectional control, in which every vehicle uses measured relative positions with its two neighboring vehicles, and every vehicle uses the same controller that is furthermore symmetric with respect to front and back. That is, the spacing error with respect to the vehicle in front has equal importance to that with respect to the vehicle behind. The noise sensitivity of this architecture was investigated earlier in [32]. This chapter answers a few questions left unanswered in [32].

We show that the stability and noise-sensitivity of this architecture depends on the number of integrators in the loop transfer function of the plant (vehicle dynamics). If there are more than two integrators, or if the vehicle dynamics is non-minimum phase, closed-loop will become unstable for a sufficiently large number of vehicles, no matter how the controller is designed. If there is a single integrator, the steady state spacing errors for a constant velocity reference will grow without bound as the number of vehicles increases, but for two integrators, the steady state errors will go to 0 for an arbitrary number of vehicles. When there are no integrators in the controller, the symmetric bidirectional architecture suffers from the “slinky effect”, namely, the measurement noise and disturbances acting on the vehicles will be amplified without bound as the number of vehicles increases. This amplification occurs whether the vehicle dynamic model has either one or two integrators.

4. In order to ameliorate some of the limitations of the symmetric bidirectional architecture, we propose a methodology to design separate controllers for ev-

ery vehicle. To handle the complexity of this design problem, which involves designing  $2N$  separate controllers (where  $N$  is the number of vehicles), we use an alternate modeling framework.

We first derive from first principles a partial differential equation (PDE) based model of the platoon dynamics, by taking a continuum approximation of the platoon when the number of vehicles is large. This approach is motivated by the extensive literature exists on PDE modeling of traffic flow. We design the control gains by a mistuning-based method, whereby the gains from the nominal, symmetric design are altered by small amounts. This mistuning design nevertheless achieves an order of magnitude improvement compared to the symmetric design. In particular, we show that the least stable eigenvalue of the closed loop platoon dynamic decays to 0 as  $O(1/N^2)$  in the symmetric bidirectional case, where  $N$  is the number of vehicles. However, the decays is only  $O(1/N)$  with the mistuning design, even with an arbitrarily small amount of mistuning. For large  $N$ , this results in an order of magnitude improvement in the stability margin. The benefits are seen to be significant even for small values of  $N$ . The predictions from the PDE analysis are corroborated by numerical calculations on the state-space representation of the platoon dynamics.

In summary of both estimation and control with relative measurements, we note that the concept of matrix-valued effective resistance for weighted graphs introduced in this dissertation is seen to be useful in both classes of problems. The effective resistance was shown to characterize the estimation error of the optimal estimates in the first part of this dissertation, and the above discussion shows that it is also relevant in the study of stability margin and noise-sensitivity of formation control algorithms.

## 1.3 Notation

A list of the notation used throughout this dissertation is provided below for easy referral. Specific notation is introduced where it is first used.

Table 1.1. Notation

$\mathbb{R}, \mathbb{N}, \mathbb{C}$	Set of real, natural and complex numbers, respectively
$I_k$	identity matrix in $\mathbb{R}^{k \times k}$
$\mathbb{S}^{k+}$	$k \times k$ symmetric positive definite matrices
$\mathcal{H}$	real Hilbert space
$X \geq 0$	$X$ is a positive semi-definite matrix
$X \succeq 0$	$X$ is entry-wise non-negative
$\mathcal{G}$	graph
$\mathcal{V}, n$ or $N$	node set, number of nodes
$\mathcal{V}_r, n_r$	set of reference nodes, number of reference nodes
$\mathcal{E}$	edge set
$\mathcal{G}^{(h)}, \mathbf{Z}_d$	$h$ -fuzz of $\mathcal{G}$ , $d$ -dimensional square lattice
$e \sim u$	edge $e$ is either $(u, v)$ or $(v, u)$ for some $v$
$A, A_b$	incidence matrix, basis incidence matrix
$\mathcal{A}$	generalized incidence matrix, $A \otimes I_k$
$\mathcal{L}$	(generalized, or matrix-weighted) graph Laplacian.
$\mathcal{D}, \mathcal{C}$	(matrix-weighted) degree matrix, adjacency matrix
$\mathcal{L}$	(matrix-weighted) Dirichlet Laplacian matrix
$\mathcal{M}, \mathcal{N}$	(matrix-weighted) basis degree matrix, basis adjacency matrix
$L, M, N$	scalar weighted versions of $\mathcal{L}, \mathcal{M}, \mathcal{N}$
$P_e$	covariance of the measurement error on edge $e \in \mathcal{E}$
$R_e$	matrix resistance on edge $e$ .
$R_{u,v}^{\text{eff}}$	generalized effective resistance between $u$ and $v$
$\Sigma_{u,o}$	error covariance of $\hat{x}_u$ 's BLUE estimate, with $o$ as the reference
$j, \mathbf{j}$	flow, flow intensity
$i, \mathbf{i}$	current, current intensity

# Part I

## Estimation with Relative Measurements

## Chapter 2

# Estimation with relative measurements: applications and the optimal estimator

In this chapter we formally describe the problem of estimation from relative measurements. First, in Section 2.1, we describe the applications – which mostly come from sensor and actuator networks – where this estimation problem is relevant. These include sensor localization, camera network calibration, time synchronization, motion coordination, and mobile robot localization. We discuss the importance of these problems, and describe in detail, in each case, how the relative measurements are obtained in practice. Then Section 2.2 formally defines the optimal estimation problem in terms of a graph, and presents the solution to the optimal estimation problem.

## 2.1 Applications

### 2.1.1 Sensor network localization

Consider a network of sensor nodes that are deployed in a large geographical area. Nodes in a sensor network are often not equipped with GPS, since they are required to be small, cheap, and are expected to operate for a long time with a battery of limited life [10]. On the other hand, in almost all potential and realized applications of sensor networks that impose these constraints, such as habitat monitoring [33], forest fire detection [34], hazardous area and perimeter surveillance, structural health monitoring [35], military reconnaissance and target tracking [36], knowledge of the nodes' locations is critical for the user of the network. The localization problem consists of estimating node locations from measurements that the sensors can provide. Although a sensor does not know its position in a global coordinate system, it can usually measure its position relative to a set of nearby nodes. These measurements can be obtained in a number of ways that depend on the sensing technology available and the application domain, which are described below.

When the sensor nodes in question are equipped with wireless devices, range measurements can be obtained by a variety of techniques, such as received signal strength [37] and time of arrival [38] measurements. In certain scenarios, they can be fitted with acoustic ranging devices [39]. Angle measurements with small form-factor devices is more challenging, though possible – albeit with limited accuracy – with switched microstrip antenna arrays [40]. Assuming that each node has a local compass to measure bearing with respect to a common North, noisy measurements of  $r_{u,v}$  and  $\theta_{u,v}$ , range and bearing, between a pair of sensors

$u$  and  $v$  are converted to noisy measurements of relative position in the  $x-y$  plane as

$$\zeta_{u,v} = \begin{bmatrix} r_{u,v} \cos \theta_{u,v} \\ r_{u,v} \sin \theta_{u,v} \end{bmatrix}.$$

The same procedure is performed for every pair of sensors that can measure their relative range and bearing. Since the range and bearing measurements have errors, the relative position  $\zeta_{u,v}$  measured in Cartesian coordinate also has error, which can be approximated as an ellipsoidal region characterized by a covariance matrix (See Figure 2.1 for a schematic). Measurement errors between distinct pairs of nodes can be assumed uncorrelated.

When the sensor nodes have on-board cameras, it is again possible to measure relative positions between pairs of nodes whose cameras have an opportunity to view a common object in an overlapping field of view (see Figure 2.2 for a schematic). Measuring the relative position and orientation between two cameras involve collaborative information gathering and processing, and is referred to as *camera network calibration*. The reader is referred to [42] and references therein for the details of obtaining such measurements. Typically, when two cameras  $u$  and  $v$  take part in calibration and exchange their local calibration parameters, one of the cameras, say  $u$ , estimate the relative position of  $v$  w.r.t. itself in its local coordinate frame, denoted by  $p_{v,u}$  (which is either a 2-vector or a 3-vector). Assuming that the rotation matrix  $T_{uo}$  that specifies the rotation from  $u$ 's local coordinate to the common global coordinate frame attached to the reference node  $o$  is available to  $u$ , it can estimate the position of node  $v$  w.r.t. itself (i.e.,  $x_v - x_u$ ) in a common Cartesian reference frame as

$$\zeta_{vu} = T_{uo} p_{vu}.$$

As long as the errors in the estimated quantities  $T_{ou}$  and  $p_{vu}$  are additive and

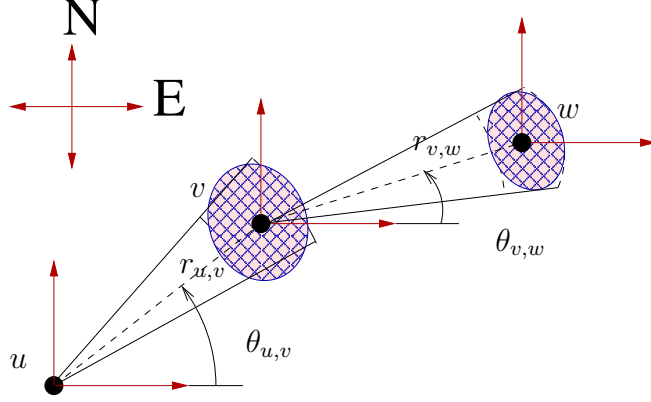


Figure 2.1. Relative position measurement between pairs of nodes in a Cartesian reference frame using range and bearing measurements. Noisy measurements of range and bearing can be converted to noisy measurements of relative position as  $\zeta_{u,v} = [r_{u,v} \cos \theta_{u,v}, r_{u,v} \sin \theta_{u,v}]^T$ . The errors in range and bearing result in an error in relative position. Although this error in general leads to a non-convex uncertainty region, it can be approximated as an ellipsoidal one (shown as the patterned region), which is characterized by a measurement error covariance matrix. An example of how the measurement noise covariance can be estimated can be found in Section 3.3.3.

zero mean, the error in this relative position measurement is also additive and zero mean. The rotation matrix  $T_{uo}$  can be estimated as the product of the rotation matrices in a path from  $o$  to  $u$  in the “graph” that describes relative calibrations. For example, if  $(u, v)$ ,  $(v, w)$ , and  $(w, o)$  are pairs of nodes such that their calibration parameters are known, then an estimate of  $T_{uo}$  is  $T_{uo} = T_{wo}T_{vw}T_{uv}$ .

In both the situations described above, two nearby sensors  $u$  and  $v$  located at positions  $p_u$  and  $p_v$ , respectively, have access to the measurement

$$\zeta_{u,v} = p_u - p_v + \epsilon_{u,v}, \epsilon_{u,v} \in \mathbb{R}^k$$

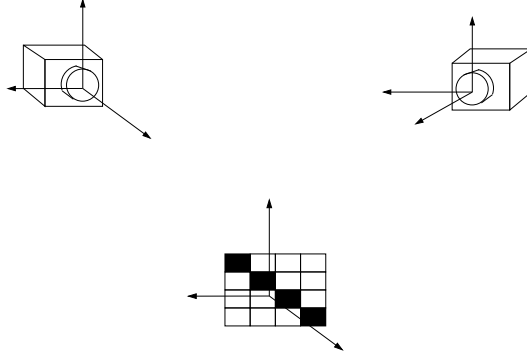


Figure 2.2. Relative position measurement between two cameras in a common Cartesian reference frame. The cameras have an overlapping field of view and have the opportunity to view an object of known size and shape. Each camera then estimates its orientation with respect to the object and its distance from the object [41]. By exchanging this information, the two cameras can estimate their relative position and orientation. See [42] for more details.

where  $\epsilon_{u,v}$  denotes measurement error. The dimension of the positions and relative measurements,  $k$ , can be either 2 or 3, or even 1 in special cases. The problem of interest is to use the  $\zeta_{u,v}$ 's to estimate the positions of all the nodes in a common coordinate system whose origin is fixed arbitrarily at one of the nodes.

We do not consider the problem of localization from range measurements alone (or angle measurements alone), on which an extensive literature exists [39, 43–50]. When only range measurements are available, the relationship between measurements and variables are non-linear. The difficulty of this non-linear problem is well-recognized, especially in the presence of noise [47]. Recognizing this difficulty, localization with both range and angle measurements is being examined recently [51, 52].

### 2.1.2 Time synchronization

Consider a set of nodes forming a multi-hop communication network, where each node has a local clock. Clocks have two sources of inaccuracy in practice: *skew* and *offset*. Skew refers to the rate at which clocks measure time and offset refers to the difference between the local times of two clocks that have the same skew. At a particular “global” time  $t$ , the measured local time at a clock can be modeled by  $\alpha + \beta t$ , where  $\alpha$  is the offset w.r.t the global time and  $\beta$  is the skew. Time synchronization consists of estimating the skews and offsets of all the nodes with respect to a common reference so that every node can read off the global time from its local clock.

Time synchronization in sensor and actuator networks is important for a number of reasons. First, sensor nodes need to coordinate and collaborate to achieve their sensing tasks. In target tracking, for example, the location of a target and its trajectory is estimated from the reports by the sensor nodes on when they sensed the target [53]. Second, to operate for a long time with limited battery power, sensor nodes typically turn off power consuming components for long periods and wake up at predetermined times [54]. Such sleep-scheduling requires a precise timing between nodes. Third, communication protocols such Time Division Multiple Access (TDMA) requires time synchronization for scheduling communication between wireless devices. Fourth, feedback control in a sensor-actuator network requires knowledge of a common time [55].

The relative skew and offset between a pair of nodes in a network can be measured (upto some error) by several methods, which are described below.

*Case A: offset without skew:* Consider first the case when all clocks have the same skew but have different offsets. All skews can be assumed to be 1 without loss

of generality. Suppose that nodes  $u$  and  $v$  in Figure 2.3 can communicate directly with each other and have clock offsets  $t_u$  and  $t_v$  with respect to a reference clock. Node  $u$  transmits a message, say, at global time  $t$ , while transmitter  $u$ 's local time is  $\tau_{tu} = t + t_u$ . The receiver  $v$  receives this message at a later time, when its local clock reads  $\tau_{rv} = t + t_v + \delta_{u,v}$ , where  $\delta_{u,v}$  is the random transmission delay from  $u$  to  $v$ . The transmission delay in fact arises from several hardware-level issues at both the transmitter and the receiver, such as the randomness in the processor load, delay in accessing the medium, and length of the messages. An extensive and excellent description of these issues can be found in [56]. Some time later, say at  $t'$  (global time), node  $v$  sends a message back to  $u$ , when its local time is  $\tau'_{tv} = t' + t_v$ . This message includes the values  $\tau_{rv}$  and  $\tau'_{tv}$  in the message body. Receiver  $u$  receives this message at local time  $\tau'_{ru} = t' + t_u + \delta_{uv}$ , where the delay  $\delta_{vu}$  has the same mean as the delay  $\delta_{uv}$ . Node  $u$  can now estimate the clock offsets as  $\zeta_{u,v} = \frac{1}{2} [(\tau'_{ru} - \tau'_{tv}) - (\tau_{rv} - \tau_{tu})] = t_u - t_v + (\delta_{vu} - \delta_{uv})/2$ . The error  $\epsilon_{u,v} := (\delta_{vu} - \delta_{uv})/2$  has zero mean as long as the delays  $\delta_{uv}$  and  $\delta_{vu}$  have the same expected value. The measured clock offset between  $u$  and  $v$  is now

$$\zeta_{u,v} = t_u - t_v + \epsilon_{u,v} \in \mathbb{R},$$

which is of the form (1.1). Similarly, the measurement of clock offsets between nodes  $v$  and  $w$  is  $\zeta_{v,w} = t_v - t_w + \epsilon_{v,w}$ . Note that for the measurement errors to be zero mean, the transmission delay between  $u$  and  $v$  has to have the same mean as the one between  $v$  and  $u$ . The task is now to estimate the clock offsets with respect to the global time, which is defined to be the local time at some reference node.

*Case B: offset and skew:* Now let us consider the general case when both clock skews and offsets are present and need to be estimated. Suppose node  $u$  transmits two messages to  $v$ , the first one at (global) time  $t_1$  and the second one at (global)

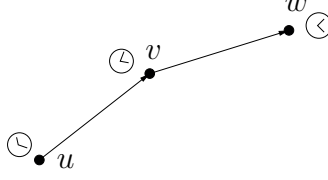


Figure 2.3. Measurement of differences in local times by bidirectional exchange time-stamped messages.

time  $t_2$ . Define  $T = t_2 - t_1$  and denote by  $\alpha_u$  and  $\alpha_v$  the clock skews of node  $u$  and  $v$  relative to a reference clock. Then the time interval of transmission recorded by  $u$  is  $m_1 := \alpha_u T$ . On the other hand, these two messages are received by node  $v$  at local times  $\alpha_v(t_1 + \delta_{uv}) + t_v$  and  $\alpha_v(t_1 + \delta'_{uv}) + t_v$ , respectively, where  $\delta_{uv}$  and  $\delta'_{uv}$  are two realizations of the random transmission delay from  $u$  to  $v$ . Node  $v$  can then compute the difference between the local reception times, which is  $m_2 := \alpha_v T(1 + \frac{\delta_{uv} - \delta'_{uv}}{T})$ . Node  $v$  then sends the measurement  $m_2$  back to node  $u$ . From the numbers  $m_1$  and  $m_2$  available to it, node  $u$  can compute the following

$$\begin{aligned} \log \frac{m_2}{m_1} &= \log \frac{\alpha_v}{\alpha_u} \left(1 + \frac{\delta_{uv} - \delta'_{uv}}{T}\right) = \log \alpha_v - \log \alpha_u + \log \left(1 + \frac{\delta_{uv} - \delta'_{uv}}{T}\right) \\ &\approx \log \alpha_v - \log \alpha_u + \frac{\delta_{uv} - \delta'_{uv}}{T}, \end{aligned}$$

under the assumption that  $\delta_{uv} - \delta'_{uv} \ll T$ . Define  $x_u := \log \alpha_u$  and  $x_v := \log \alpha_v$  and  $\epsilon_{uv} := \frac{\delta_{uv} - \delta'_{uv}}{T}$ . Then the above measurement is equivalent to

$$\zeta_{vu} := \log \frac{m_2}{m_1} = x_v - x_u + \epsilon_{vu},$$

which is of the form (1.1). Note that  $\epsilon_{uv}$  is zero-mean random variable as long as  $\delta_{uv}$  and  $\delta'_{uv}$  have the same expected value. In this way, noisy relative skews can be measured between pairs of nodes that can exchange time-stamped messages. To estimate the offset, one first scales the local time of one of the sensors with the already estimated relative skew, so that in the new time coordinate, the only

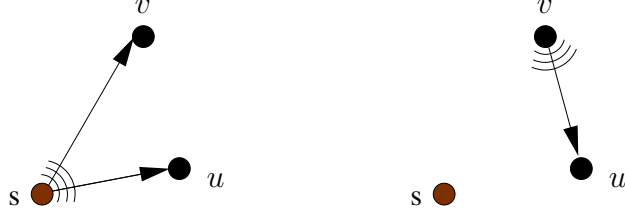


Figure 2.4. RBS: measurement of differences in local times by unidirectional message exchange [1]. Both  $u$  and  $v$  receives a message transmitted by source  $s$  at the same time. The local time at  $u$  when it received this message is later broadcast. Upon receiving this message, node  $v$  can measure its relative offset with  $u$ . Communication from  $v$  to  $u$  is not needed. So RBS allows relative offset measurements even when communication is asymmetric.

difference between the local times is an offset. This offset can then be measured as described earlier. At the end, it is scaled back to obtain an noisy estimate of the true offset. This way, both relative skews and offsets are measured that follow the measurement model (1.1).

All the measurement techniques are, however, susceptible to measurement errors, especially when wireless communication is involved. Therefore the error term  $\epsilon_e$  on a measurement  $\zeta_e$  has to be paid careful attention to. These errors come from several hardware-level issues at the both transmitter and receiver, such as the randomness in the processor load, delay in accessing the medium, and length of the messages. An extensive description of these issues can be found in [56].

The measurement techniques outlined above require bidirectional message exchange between pairs of nodes. Measurements of relative skews and offsets between a pair of nodes can also be obtained by the RBS(Reference Broadcast System) method, which does not require bidirectional message exchange between them but requires the involvement of a third node [1]. The RBS method is ex-

plained in Figure 2.4 briefly. Node  $p$  broadcasts a message marked as a synchronization signal to its neighbors, and since time of propagation of radio waves is negligible compared to the other sources of delay in transmission and reception, nodes  $u$  and  $v$  receive the message at the same global time, say  $t$ . From the time the messages is processed by the receiver antenna and physical layer of the wireless device to the time it arrives at the application layer of the protocol stack where the local time can be recorded, there will be a processing delay. Therefore, the local times at nodes  $u$  and  $v$  recorded as the time of reception of the same message are  $\tau_{ru} := t + t_u^p + t_u$  and  $\tau_{rv} := t + t_v^p + t_v$ , respectively, where  $t_u^p, t_v^p$  are receiver-side processing delays at  $u$  and  $v$ . Node  $u$  then sends the recorded receive time to  $v$  (or vice versa), and  $v$  can estimate the difference between their clock offsets:

$$\zeta_{v,u} = \tau_{rv} - \tau_{ru} = t_v - t_u + (t_v^p - t_u^p) = t_v - t_u + \epsilon_{v,u},$$

where  $\epsilon_{v,u} := t_v^p - t_u^p$  is zero-mean as long as the processing delays at both the receivers  $u$  and  $v$  have the same expected value. It should be stressed that such a relative measurement is available to  $v$  as long as it can receive messages from  $u$ , even if  $u$  is unable to receive messages from  $v$ .

### 2.1.3 Motion Consensus

Consider the situation where, in a group consisting of several mobile agents, each agent wants to determine its velocity with respect to the velocity of a leader using only measurements of its relative velocities with respect to nearby agents. These measurements can be obtained, for example, by using vision-based sensors. In particular, two nearby agents  $u$  and  $v$  moving with velocities  $\dot{p}_u$  and  $\dot{p}_v$ ,

respectively, have access to the measurement

$$\zeta_{u,v} = \dot{p}_u - \dot{p}_v + \epsilon_{u,v},$$

where  $\epsilon_{u,v}$  denotes measurement error. The task is to determine the velocity of each agent with respect to the leader based solely on the available relative velocities between pairs of neighboring agents. The same problem arises when the agents are trying to estimate their headings with respect to that of a leader using noisy measurements of relative headings between certain pairs of agents.

A similar situation arises in a group of mobile nodes when pairs of nearby nodes can measure their relative heading, and each agent wants to estimate its heading with respect to a leader.

#### 2.1.4 Mobile robot localization

Consider a group of mobile robots such that certain pairs of robots can measure their relative positions periodically, which can be obtained either by vision-based techniques [57] or by a stereo-ranging device [58], or by one of the techniques explained in the previous section. Furthermore, each robot can measure how much it has moved in a given time interval, which can be obtained by dead-reckoning [59]. The problem is to measure the position of each robot at the current time based on all the relative position measurement.

Figure 2.5 depicts such a situation schematically. Two types of relative position measurements are available: (i) those between two distinct robots at the same time instant:  $\zeta_{u_t,v_t} = x_{u_t} - x_{v_t} + \epsilon_{u_t,v_t}$ , and (ii) those between the same robot at two consecutive time instants:  $\zeta_{u_{t+1},u_t} = x_{u_{t+1}} - x_{u_t} + \epsilon_{u_{t+1},u_t}$ . In terms of the estimation problems described above, the robot positions at various times can be

thought of as variables to be estimated. Since only the initial position of one robot is assumed known, the number of variables to be estimated grows with time.

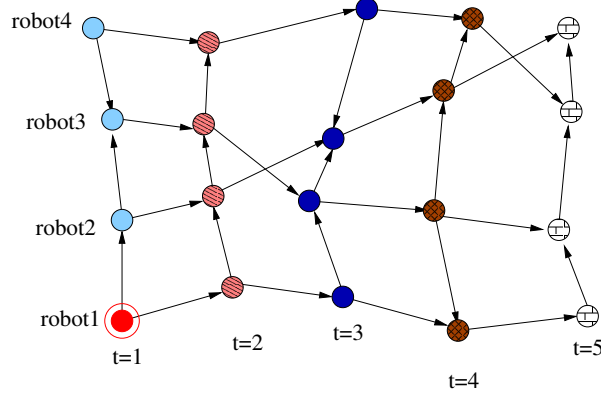


Figure 2.5. A group of mobile robots. Snapshots of the positions of four robots at five different time instants are shown schematically. The unknown variables are the robot positions  $x_u(t)$ ,  $u = 0, 1, \dots, N$ , where  $N$  is the number of robots, at the time instants  $t = 1, 2, \dots, T$ , where  $T$  is the current time index, except for the initial position of robot 1:  $x_1(0)$ , which is taken as the reference. Two types of relative position measurements are available: (i) those between two distinct robots at the same time instant -  $\zeta_{u_t, v_t} = x_{u_t} - x_{v_t} + \epsilon_{u_t, v_t}$ , and (ii) those between the same robot at two consecutive time instants  $\zeta_{u_{t+1}, u_t} = x_{u_{t+1}} - x_{u_t} + \epsilon_{u_{t+1}, u_t}$ . Since only the initial position of one robot is assumed known, the number of variables to be estimated grows with time. In the terminology of Section 2.2, the measurement graph is a function of time.

## 2.2 Measurement graph and optimal estimation

The estimation problem can be posed in terms of a directed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  whose vertices or nodes represent variables and whose edges represent noisy relative measurements. That is, for every  $e \in \mathcal{E}$ , a measurement of the following form is available:

$$\zeta_e = x_u - x_v + \epsilon_e, \quad \forall e = (u, v) \in \mathcal{E}, \quad u, v \in \mathcal{V}. \quad (2.1)$$

where  $\epsilon_e$  is measurement error. The graph  $\mathcal{G}$  is called the *measurement graph*. In the sequel, we use the symbol  $e$  to denote not only an edge  $(u, v)$  but also the index of the edge as well, so that  $e$  can take values in the set  $\{1, 2, \dots, m\}$ , where  $m$  is the total number of measurements. The covariance of the measurement error  $\epsilon_e$  is  $P_e := \mathbb{E}[\epsilon_e \epsilon_e^T]$ . The measurement error covariances  $P_e$ ,  $e \in \mathcal{E}$  are assumed known. The measurement errors on distinct edges are assumed uncorrelated, i.e.,  $\mathbb{E}[\epsilon_e \epsilon_{\bar{e}}^T] = 0$  if  $e \neq \bar{e}$ . With relative measurements alone, determining  $x_u$  is possible only up to an additive constant. To avoid this ambiguity, we assume that at least one of the nodes is used as a reference by all of the nodes, and therefore its node variable can be assumed known. When several node variables are known, we can have several references. The set of *reference nodes* is denoted by  $\mathcal{V}_r$ , where  $\mathcal{V}_r \subset \mathcal{V}$ . An edge  $e = (u, v)$  is said to be incident on the nodes  $u$  and  $v$ . We write  $e \sim u$  to denote that  $e$  is incident on  $u$ .

Depending on the application, the measurement graph can vary with time. In the sensor localization, time synchronization, and motion consensus problems described earlier, it was implicitly assumed that the variables are fixed and measurements are obtained once. As a result, the measurement graph was time-invariant. However, in problems such mobile robot localization, new variables and measurements appear over time, and consequently the measurement graph

is time-varying. When the application leads to a time-varying graph, we can sometimes examine the graph obtained by collecting all the variables and measurements over a time-period. In this dissertation we only consider measurement graphs that are time-invariant.

### 2.2.1 The optimal estimator (BLUE) and the optimal estimates

The task is to estimate all of the unknown node variables from the measurements and the reference variables. We examine the optimal estimate of the node variables. The optimal estimates refer to the ones obtained by the best linear unbiased estimator (BLUE), which has the minimum variance among all linear estimators [9].

Consider a measurement graph  $\mathcal{G}$  with  $n$  nodes and  $m$  edges. Recall that  $k$  is the dimension of the node variables. Let  $\mathbf{X}$  be a vector in  $\mathbb{R}^{nk}$  obtained by stacking together all the node variables, known and unknown, i.e.,  $\mathbf{X} := [x_1^T, x_2^T, \dots, x_n^T]^T$ . Define  $\mathbf{z} := [\zeta_1^T, \zeta_2^T, \dots, \zeta_m^T]^T \in \mathbb{R}^{km}$  and  $\boldsymbol{\epsilon} := [\epsilon_1^T, \epsilon_2^T, \dots, \epsilon_m^T]^T \in \mathbb{R}^{km}$ . This stacking together of variables allows us to rewrite (2.1) in the following form:

$$\mathbf{z} = \mathcal{A}^T \mathbf{X} + \boldsymbol{\epsilon}, \quad (2.2)$$

where  $\mathcal{A}$  is a matrix uniquely determined by the graph. To construct  $\mathcal{A}$ , we start by defining the *incidence matrix*  $A$  of the graph  $\mathcal{G}$ , which is an  $n \times m$  matrix with one row per node and one column per edge defined by  $A := [a_{ue}]$ , where  $a_{ue}$  is nonzero if and only if the edge  $e \in \mathcal{E}$  is incident on the node  $u \in \mathcal{V}$  [60]. When nonzero,  $a_{ue} = -1$  if the edge  $e$  is directed towards  $u$  and  $a_{ue} = 1$  otherwise. The matrix  $\mathcal{A}$  that appears in (2.2) is an expanded version of the incidence matrix  $A$ ,

defined by

$$\mathcal{A} := A \otimes I_k, \quad (2.3)$$

where  $I_k$  is the  $k \times k$  identity matrix and  $\otimes$  denotes the Kronecker product. Essentially, every entry of  $A$  is replaced by a matrix of the form  $a_{ue}I_k$  to construct the matrix  $\mathcal{A}$  (see Figure 2.6 for an example). We call  $\mathcal{A}$  the *generalized incidence matrix of  $\mathcal{G}$* .

Let  $n_r$  denote the number of reference variables and  $n_b$  denote the number of unknown variables. Let  $A_b$  be the submatrix of  $A$  that is obtained by removing those rows from  $A$  that correspond to the reference nodes in  $\mathcal{V}_r$ , so that it contains only those rows of  $A$  that correspond to the nodes in  $\mathcal{V} \setminus \mathcal{V}_r$ . The matrix  $A_b$  is called the *basis incidence matrix* [61]. Clearly, the incidence matrix can be decomposed as

$$A = \begin{bmatrix} A_b \\ A_r \end{bmatrix},$$

where  $A_b \in \mathbb{R}^{n_b \times m}$  and  $A_r \in \mathbb{R}^{n_r \times m}$ . Define

$$\mathcal{A}_b := A_b \otimes I_k, \quad \mathcal{A}_r := A_r \otimes I_k,$$

where  $\mathcal{A}_b$  is now termed *generalized basis incidence matrix of  $\mathcal{G}$  with reference node set  $\mathcal{V}_r$* .

By partitioning  $\mathbf{X}$  into a vector  $\mathbf{x} \in \mathbb{R}^{kn_b}$  containing all the unknown node variables and another vector  $\mathbf{x}_r \in \mathbb{R}^{kn_r}$  containing all the known reference node variables:  $\mathbf{X}^T = [\mathbf{x}_r^T, \mathbf{x}^T]^T$ , we can re-write (2.2) as

$$\mathbf{z} = \mathcal{A}_r^T \mathbf{x}_r + \mathcal{A}_b^T \mathbf{x} + \boldsymbol{\epsilon},$$

where  $\mathcal{A}_r$  contains the rows of  $\mathcal{A}$  corresponding to the reference nodes and  $\mathcal{A}_b$  contains the rows of  $\mathcal{A}$  corresponding to the unknown node variables. The equation

above can be further rewritten as:

$$\bar{\mathbf{z}} = \mathcal{A}_b^T \mathbf{x} + \boldsymbol{\epsilon}, \quad (2.4)$$

where  $\bar{\mathbf{z}} := \mathbf{z} - \mathcal{A}_r^T \mathbf{x}_r$  is a known vector.

Estimation of the unknown node variables in the vector  $\mathbf{x}$  based on the linear measurement model (2.4) is a classical estimation problem. Since  $\boldsymbol{\epsilon}$  is a random vector with zero mean and covariance matrix

$$\mathcal{P} := \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T], \quad (2.5)$$

the BLU estimate  $\hat{\mathbf{x}}^*$  of  $\mathbf{x}$  is the solution to the system of linear equations [9]

$$\mathcal{L}\hat{\mathbf{x}}^* = \mathbf{b}, \quad (2.6)$$

where

$$\mathcal{L} := \mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T \quad (2.7)$$

$$\mathbf{b} := \mathcal{A}_b \mathcal{P}^{-1} (\mathbf{z} - \mathcal{A}_r^T \mathbf{x}_r). \quad (2.8)$$

Since the measurement errors on two different edges are uncorrelated,  $\mathcal{P}$  is a symmetric positive definite block diagonal matrix with the measurement error covariances along the diagonal:  $\mathcal{P} = \text{diag}(P_1, P_2, \dots, P_m) \in \mathbb{R}^{km \times km}$ , where  $P_e$  is the covariance of the measurement error  $\epsilon_e$ .

The next theorem establishes necessary and sufficient conditions on the measurement graph  $\mathcal{G}$  so that the optimal estimate of node variables is unique and shows how the covariance of the estimation error  $\mathbf{x} - \hat{\mathbf{x}}^*$  relates to the matrices associated with the graph  $\mathcal{G}$ . The existence and uniqueness condition is one of weak connectivity of the directed graph  $\mathcal{G}$ . A directed graph is *weakly connected*

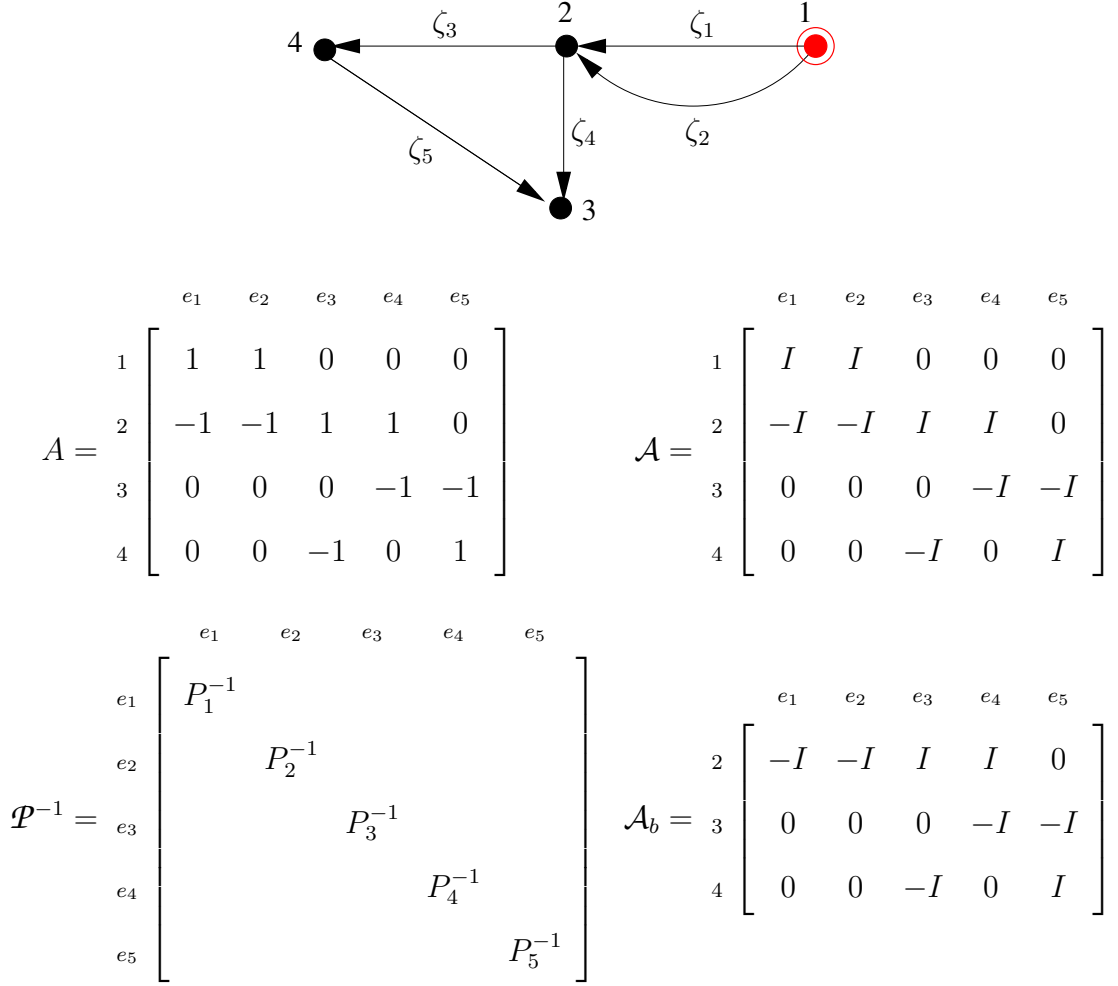


Figure 2.6. A measurement graph  $\mathcal{G}$  and a few of its associated matrices: the incidence matrix  $A$ , the generalized incidence matrix  $\mathcal{A}$ , the generalized basis incidence matrix  $\mathcal{A}_b$ , and the edge-covariance matrix  $\mathcal{P}$ . The row and column indices of  $A$  correspond to node and edge indices, respectively. The single positive entry in each column of  $A$ , namely 1, indicates the start node of the corresponding edge in  $\mathcal{G}$ , while the single negative entry  $-1$  indicates the end node.

if it is possible to go from every node to every other node of the graph traversing the edges, not necessarily respecting the edge directions. An equivalent definition is that it is weakly connected when there is an *undirected path* between every pair

of nodes. An undirected path  $\mathcal{P}$  from a node  $p_1$  to another node  $p_m$  in a graph  $\mathcal{G}$  is an alternating sequence of finite number of nodes and edges that start with  $p_1$  and end with  $p_m$ :

$$\mathcal{P} = \{p_1, e_1, p_2, e_2, \dots, p_l, e_l, p_{l+1}, \dots, p_{m-1}, p_m\}$$

such that every edge  $e_l$  in the path is incident on the nodes  $p_l, p_{l+1}$  adjacent to it in the path, and no edges or nodes are repeated. If a graph is not weakly connected, it can be decomposed into a number of disjoint subgraphs such that every one of them is weakly connected. These subgraphs are called connected components of the graph. When every weakly connected component of the measurement graph has at least one reference node, we say that the graph is *weakly connected to*  $\mathcal{V}_r$ , the set of reference nodes.

We call the pair  $(\mathcal{G}, P)$ , where  $\mathcal{G}$  is a measurement graph and  $P : \mathcal{E} \rightarrow \mathbb{S}^{k+}$  is a function that assigns measurement error covariances to the edges of the graph, a *measurement network*.

The next theorem establishes conditions for the well-posed-ness of the BLUE estimation problem, whose proof is provided in Section 2.3.

**Theorem 2.2.1.** *The matrix  $\mathcal{L}$  defined in (2.7) for a finite measurement network  $(\mathcal{G}, P)$  is invertible, and therefore BLU estimate  $\hat{\mathbf{x}}^*$  exists and is unique, if and only*

$$\mathcal{L} = \mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T = \begin{matrix} & \begin{matrix} 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} P_1^{-1} + P_2^{-1} + P_3^{-1} + P_4^{-1} & -P_3 & -P_4^{-1} \\ -P_4^{-1} & P_4^{-1} + P_5^{-1} & -P_5^{-1} \\ -P_3^{-1} & -P_5^{-1} & P_3^{-1} + P_5^{-1} \end{bmatrix} \end{matrix}$$

Figure 2.7. Dirichlet Laplacian for the graph in Figure 2.6.

Consider the measurement graph  $\mathcal{G}$  with 4 nodes and 5 edges shown in Figure 2.6. Node 1 is the reference. The incidence matrix  $A$  is therefore a  $4 \times 5$  matrix consisting of 0s, 1s, and  $-1$ s. The matrix form (2.2) of the measurement equations (1.1) for this graph is

$$\underbrace{\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \\ \zeta_5 \end{bmatrix}}_{\mathbf{z}} = \underbrace{\begin{bmatrix} I & -I & 0 & 0 \\ I & -I & 0 & 0 \\ 0 & I & 0 & -I \\ 0 & I & -I & 0 \\ 0 & 0 & -I & I \end{bmatrix}}_{\mathcal{A}^T} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\mathbf{X}} + \underbrace{\begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \end{bmatrix}}_{\boldsymbol{\epsilon}},$$

where  $I$  is the  $k \times k$  identity matrix. The 4 node variables in the vector  $\mathbf{X}$  are related to the 5 measurements in the vector  $\mathbf{z}$  by the  $4k \times 5k$  matrix  $\mathcal{A}$ , the expanded version of the incidence matrix. The measurement model (2.4) when node 1 is the reference with  $x_1 = 0$  is

$$\underbrace{\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \\ \zeta_5 \end{bmatrix}}_{\mathbf{z}} = \underbrace{\begin{bmatrix} I \\ I \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathcal{A}_r^T} \underbrace{0}_{\mathbf{x}_r} + \underbrace{\begin{bmatrix} -I & 0 & 0 \\ -I & 0 & 0 \\ I & 0 & -I \\ I & -I & 0 \\ 0 & -I & I \end{bmatrix}}_{\mathcal{A}_b^T} \underbrace{\begin{bmatrix} x_2 \\ x_3 \\ x_4 \end{bmatrix}}_{\mathbf{x}} + \boldsymbol{\epsilon}.$$

The relationship between the 3 unknown node variables in the vector  $\mathbf{x}$  are related to the known quantities, that is, measurements  $\mathbf{z}$  and the reference variable  $x_1$ , by the  $3k \times 5k$  matrix  $\mathcal{A}_b$ .

Figure 2.8. An example of a measurement graph, the BLUE estimates and error covariances.

*if the measurement graph  $\mathcal{G}$  is weakly connected to its reference nodes  $\mathcal{V}_r$ . When  $\mathcal{L}$  is non-singular, the estimation error covariance matrix  $\boldsymbol{\Sigma} := \mathbb{E}[(\mathbf{x} - \hat{\mathbf{x}}^*)(\mathbf{x} - \hat{\mathbf{x}}^*)^T]$  is given by*

$$\boldsymbol{\Sigma} = \mathcal{L}^{-1}. \quad \square$$

The covariance matrix  $\Sigma_{u,o}$  for the estimation error of a particular node vari-

(continued from Figure 2.8) Since the graph  $\mathcal{G}$  is weakly connected,  $\mathcal{L}$  is invertible. The optimal estimate of the vector  $\mathbf{x}$ , the solution to (2.6), is given by  $\hat{\mathbf{x}}^* = \mathcal{L}^{-1} \mathcal{A}_b \mathcal{P}^{-1} \bar{\mathbf{z}}$ . From Figure 2.8, it follows that the optimal estimate  $\hat{\mathbf{x}}^*$  when all measurement covariance matrices are equal to the identity matrix is

$$\begin{bmatrix} \hat{x}_2^* \\ \hat{x}_3^* \\ \hat{x}_4^* \end{bmatrix} = \underbrace{\begin{bmatrix} 4I & -I & -I \\ -I & 2I & -I \\ -I & -I & 2I \end{bmatrix}^{-1}}_{\mathcal{L}} \underbrace{\begin{bmatrix} -I & -I & 0 & 0 & 0 \\ 0 & 0 & 0 & -I & -I \\ 0 & 0 & -I & 0 & I \end{bmatrix}}_{\mathcal{A}_b \mathcal{P}^{-1}} \underbrace{\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \\ \zeta_5 \end{bmatrix}}_{\mathbf{z} - \mathcal{A}_r^T \mathbf{x}_r}$$

Note the Laplacian-like structure of the matrix  $\mathcal{L}$ . The covariance matrices of the overall estimation error and of the individual node-variable errors are

$$\Sigma = \frac{1}{6} \underbrace{\begin{bmatrix} 3I & 3I & 3I \\ 3I & 7I & 5I \\ 3I & 5I & 7I \end{bmatrix}}_{\mathcal{L}^{-1}}, \quad \Sigma_2 = \frac{1}{2}I, \quad \Sigma_3 = \frac{7}{6}I, \quad \Sigma_4 = \frac{7}{6}I.$$

The covariance of the estimation error of node  $u$  is simply the  $(u-1)^{\text{th}}$  diagonal block of the covariance matrix  $\Sigma$ .

Figure 2.9. Figure 2.8 contd.

able  $x_u$  appears in the corresponding  $k \times k$  diagonal block of  $\Sigma$ . A measurement graph, along with the corresponding measurement equations (2.2) and (2.4) and the node variable estimates computed from (2.6), is shown in figure 2.8.

Weak connectivity of the measurement graph is required not only for the existence of the optimal estimator of a node variable  $x_u$ , but also for the existence of *any unbiased* estimator of  $x_u$ . Before stating it formally, we emphasize the distinction between an estimate and an estimator. Recall that a linear estimate of a node variable is a linear combination of the measurements  $\zeta_e, e \in \mathcal{E}$  specified by a set of coefficient matrices. In particular, an estimate  $\hat{x}_u$  of a node variable

$x_u$  is given by

$$\hat{x}_u = \sum_{e \in \mathcal{E}} C_e^T \zeta_e, \quad (2.9)$$

where the function  $C : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  specifies the coefficients of the measurements. In the equation above, and in the sequel, for a function  $f$  with the edge set  $\mathcal{E}$  as the domain, we use  $f_e$  to denote the value of the function at an edge  $e \in \mathcal{E}$ . We call the function  $C$  the *estimator* of  $x_u$ . It should be stressed that the edge set  $\mathcal{E}$  is implicitly assumed to be finite. Otherwise the summation in (2.9) will be a series and more care has to be exercised in defining an estimate or an estimator. Estimation in infinite graphs is considered in Chapter 4.

Now we formally state the result on the importance of weak connectivity.

**Lemma 2.2.1.** *For a finite measurement graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with a reference node  $\mathcal{V}_r \subset \mathcal{V}$ , there exists an unbiased estimator for every node variable  $x_u$ ,  $u \in \mathcal{V} \setminus \mathcal{V}_r$  if and only if  $\mathcal{G}$  is weakly connected to  $\mathcal{V}_r$ .  $\square$*

To prove this result, we will need the concept of a flow in a graph. A *generalized flow from node  $u \in \mathcal{V}$  to node  $v \in \mathcal{V}$  with intensity  $\mathbf{j} \in \mathbb{R}^{k \times k}$*  is an edge-function  $j : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  such that

$$\sum_{\substack{(p,q) \in \mathcal{E} \\ p=\bar{p}}} j_{p,q} - \sum_{\substack{(q,p) \in \mathcal{E} \\ p=\bar{p}}} j_{q,p} = \begin{cases} \mathbf{j} & \bar{p} = u \\ -\mathbf{j} & \bar{p} = v \\ \mathbf{0} & \text{otherwise} \end{cases} \quad \forall \bar{p} \in \mathcal{V}. \quad (2.10)$$

The reason flows are useful in the analysis of the estimation problem under study is that they precisely characterize unbiased estimators, which is stated in the next lemma. The proof of this lemma is provided in Section 2.3.

**Lemma 2.2.2 (Unbiased Estimator).** *In a finite measurement network  $(\mathcal{G}, P)$  with a reference node  $o \in \mathcal{V}$ , i.e.,  $\mathcal{V}_r = \{o\}$ , an edge function  $j$  is a linear unbiased estimator of a node variable  $x_u$  if and only if  $j$  is a flow of intensity  $I_k$  from node  $u$  to the reference node  $o$ . In this case, the covariance of the error in the estimate  $\hat{x}_u$  is given by*

$$\mathbb{E}[(x - \hat{x}_u)(x - \hat{x}_u)^T] = \sum_{e \in \mathcal{E}} j_e^T P_e j_e. \quad \square$$

The proof of Lemma 2.2.1, which is based on the result above, is presented in Section 2.3. Finally, the next proposition formally relates the unbiased estimators of  $x_u$  and the best linear unbiased estimator of  $x_u$ .

**Proposition 2.2.1.** *In a finite measurement network  $(\mathcal{G}, P)$  with a reference node  $o \in \mathcal{V}$ , for every node variable  $x_u$ , the best linear unbiased estimator  $C$  is the flow  $C : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  of intensity  $I_k$  from  $u$  to  $o$  that minimizes the quadratic cost*

$$\text{trace}\left(\sum_{e \in \mathcal{E}} j_e^T P_e j_e\right)$$

*among all flows  $j$  of intensity  $I_k$  from  $u$  to  $o$ .*  $\square$

The proof follows from the characterization of the BLUE as the unbiased estimator that minimizes the sum of the variances of the estimation errors [62] and the preceding discussion.

## 2.2.2 Dirichlet Laplacian and BLUE

The matrix  $\mathcal{L}$  has a structure similar to the Laplacian matrix  $L$  of the graph  $\mathcal{G}$ , which is defined as  $L := AA^T$  [60]. To explore this connection, we first consider a *matrix-weighted graph*  $\mathcal{G}$  whose edges have matrix-valued weights (that are symmetric positive definite) associated with them, specified by a function

$W : \mathcal{E} \rightarrow \mathbb{S}^{k+}$ . The symbol  $\mathbb{S}^{k+}$  denotes the set of  $k \times k$  symmetric positive definite matrices. For a matrix-weighted graph  $\mathcal{G}$  with weight function  $W$ , we define the *generalized, or matrix-weighted, graph Laplacian* as

$$\mathcal{L} := \mathcal{A} \mathcal{W} \mathcal{A}^T \in \mathbb{R}^{kn \times kn}, \quad (2.11)$$

where  $\mathcal{A}$  is the generalized incidence matrix of  $\mathcal{G}$  and  $\mathcal{W}$  is a block-diagonal matrix with edge weights on its diagonal:  $\mathcal{W} = \text{diag}(W_1, \dots, W_m)$ .

Expanding (2.11), we get

$$\begin{aligned} \mathcal{L} &= \mathcal{A} \mathcal{W} \mathcal{A}^T = \begin{bmatrix} \mathcal{A}_b \\ \mathcal{A}_r \end{bmatrix} \mathcal{W} \begin{bmatrix} \mathcal{A}_b^T & \mathcal{A}_r^T \end{bmatrix} \\ &= \left[ \begin{array}{c|c} \mathcal{A}_b \mathcal{W} \mathcal{A}_b^T & \mathcal{A}_b \mathcal{W} \mathcal{A}_r^T \\ \hline \mathcal{A}_r \mathcal{W} \mathcal{A}_b^T & \mathcal{A}_r \mathcal{W} \mathcal{A}_r^T \end{array} \right]. \end{aligned}$$

For a measurement graph if we assign edge weights as the inverses of measurement error covariances, i.e.,  $W_e = P_e^{-1}$  for every  $e \in \mathcal{E}$ , then  $\mathcal{W} = \mathcal{P}^{-1}$  and  $\mathcal{L} = \mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T$ . So  $\mathcal{L}$  is a principal submatrix of the generalized Laplacian  $\mathcal{L}$ . We call  $\mathcal{L}$  the *generalized Dirichlet Laplacian* or the *generalized grounded Laplacian* of the matrix-weighted graph  $\mathcal{G}$  with weight function  $W$  and *boundary*  $\mathcal{V}_r$ . We will frequently refer to  $\mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T$  (and  $\mathcal{A} \mathcal{P}^{-1} \mathcal{A}^T$ ) as the Dirichlet Laplacian (and Laplacian) for the *network*  $(\mathcal{G}, P)$ .

Principal submatrices of the usual graph Laplacian matrix are called Dirichlet Laplacians since they appear in the numerical solution of PDEs with Dirichlet boundary conditions. They also appear in electrical network analysis when the potential of one or more of the nodes is fixed at 0, hence they are also called grounded Laplacians. In fact, we will shortly see that  $\mathcal{L}$  plays a key role in a abstract, generalized electrical network with matrix valued currents and voltages.

We list below a few properties of the generalized Laplacian and the incidence matrix that will be used in establishing certain results later in the paper.

**Proposition 2.2.2.** *Let  $\mathcal{G}$  be a measurement graph,  $A$  its incidence matrix and  $A_b$  be the basis incidence matrix constructed by removing the rows corresponding to the reference nodes from  $A$ . Then, the following statements are true:*

1. *If  $\mathcal{G}$  is weakly connected and has  $n$  nodes, then the rank of  $A$  is  $n - 1$ , and  $\mathbf{1}^T A = 0$ , where  $\mathbf{1} \in \mathbb{R}^n$  is a vector of all 1's.*
2. *The basis incidence matrix  $A_b$  has full row rank if and only if every weakly connected component of the graph has at least one reference node.*
3.  *$\mathcal{L}$  has at least  $k$  zero eigenvalues. It has exactly  $k$  zero eigenvalues if and only if  $\mathcal{G}$  is weakly connected.*
4.  *$\mathcal{L} \times (\mathbf{1} \otimes I_k) = 0$ .* □

The last two statements are direct consequences of the first two, whose proofs are contained in the proof of Theorem 2.2.1.

*Remark 2.2.1 (Role of edge directions).* Note that the graph Laplacian  $L = AA^T$  does not depend on the directions of the edges [60]. Since  $\mathcal{P}$  is also independent of the edge-directions, it follows from the definition (2.11) that the generalized Laplacian  $\mathcal{L}$  is also independent of the edge directions chosen. Clearly, the matrix  $\mathcal{L}$ , being a submatrix of  $\mathcal{L}$ , shares this property, too. Since the BLUE error covariances are given by the inverse of  $\mathcal{L}$  (cf. Theorem 2.2.1), they do not depend on the edge directions. Therefore, as long as we are interested only in the BLUE covariances, we can regard the measurement graph as undirected. However, the optimal estimator of a node variable does depend on the edge directions. □

### 2.2.3 Role of parallel edges

Two edges  $e_1$  and  $e_2$  are said to be *parallel* if they are incident on the same set of nodes (irrespective of the direction of the edges). Parallel edges may be present in a measurement graph, e.g., when a measurement  $\zeta_{u,v}$  is obtained by one of the nodes between  $u$  and  $v$ , and the measurement  $\zeta_{v,u}$  is obtained by the other node. Such a situation can occur when two nodes measure each other's relative position by measuring range and angle, as described in Section 2.1.1. Parallel edges may also appear when multiple relative measurements between the same pair of nodes are obtained over time, and all these measurements are used to define the measurement graph.

However, a measurement network with parallel edges can be reduced to one without parallel edges, by replacing relative measurements on parallel edges with a single measurement of appropriate covariance, so that the BLU estimates and their associated error covariances don't change. Imagine there are  $\ell$  parallel edges  $e_1, e_2, \dots, e_\ell$  between a pair of nodes  $u$  and  $v$ , with associated measurements  $\zeta_1, \dots, \zeta_\ell$ , and covariances  $P_1, P_2, \dots, P_\ell$ . We can replace these  $\ell$  parallel edges with a single edge  $e' := (u, v)$  with associated measurement  $\zeta_{e'}$  and measurement error covariance  $P_{e'}$  that are given by

$$\begin{aligned} P_{e'}^{-1} &:= P_1^{-1} + \dots + P_\ell^{-1}, \\ \zeta_{e'} &:= (P_1^{-1} + \dots + P_\ell^{-1})^{-1} (s(1, e')P_1^{-1}\zeta_1 + \dots + s(\ell, e')P_\ell^{-1}\zeta_\ell), \end{aligned} \tag{2.12}$$

where  $s(e, e') = +1$  if the *orientations* of  $e$  and  $e'$  are the same, and  $s(e, e') = -1$  if the orientations are opposite. Two parallel edges  $e_1$  and  $e_2$  are said to have the same orientation if both are directed away from the same node, otherwise their orientations are opposite. Replacing a set of parallel edges by a single edge according to this procedure leaves the BLU estimates and the BLUE covariances

of all the node variables invariant. One could prove this fact by straightforward but tedious manipulations; so a formal proof is omitted. Therefore, without any loss of generality, we can assume that a measurement graph does not contain any parallel edges.

## 2.3 Proofs

*Proof of Theorem 2.2.1.* We will first consider the case when  $\mathcal{G}$  has only one connected component and prove that  $\mathcal{L}$  is invertible if and only if the graph has at least one reference node. When  $\mathcal{G}$  is weakly connected, the rank of its incidence matrix  $\text{rank}(A) = n - 1$ , where  $n$  is the number of nodes [61]. If  $\mathcal{G}$  has no reference nodes, then  $A = A_b$ , which makes  $A_b$ , and thereby  $\mathcal{A}_b$ , rank deficient. Then  $\mathcal{L} = \mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b$  is singular. On the other hand, any sub matrix obtained from  $A$  by removing one or more rows has full row rank [61]. Any smaller submatrix must obviously be full row rank. Now if the weakly connected graph  $\mathcal{G}$  has at least one reference node,  $A_b$ , is full row rank by the previous argument, and so is  $\mathcal{A}_b$ . To prove that  $\mathcal{L}$  is non-singular, assume that  $\exists x \neq 0$  s.t.  $x^T (\mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T) x = 0$ . Since  $\mathcal{P}$  is symmetric positive definite, this implies  $\mathcal{P}^{-1/2} \mathcal{A}_b^T x = 0$ , where  $\mathcal{P}^{-1/2}$  is the unique positive definite square root of  $\mathcal{P}^{-1}$ . Therefore  $\mathcal{A}_b^T x = 0$ , which is a contradiction. This proves that when  $\mathcal{G}$  is weakly connected,  $\mathcal{L}$  is invertible if and only if there is at least one reference node.

To examine the situation when  $\mathcal{G}$  has more than one weakly connected components, assume w.l.o.g. that it has two components  $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$  and  $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ . Since the two components cannot have an edge or a node in common,

the generalized incidence matrix  $\mathcal{A}$  of  $\mathcal{G}$  can be written as

$$\mathcal{A} = \begin{bmatrix} \mathcal{A}_1 & 0 \\ 0 & \mathcal{A}_2 \end{bmatrix},$$

where  $\mathcal{A}_i$  is the generalized incidence matrix of the component  $\mathcal{G}_i$ . Similarly,

$$\mathcal{A}_b = \begin{bmatrix} \mathcal{A}_{1,b} & 0 \\ 0 & \mathcal{A}_{2,b} \end{bmatrix},$$

where  $\mathcal{A}_{i,b}$  correspond to the component  $\mathcal{G}_i$ . As a result the matrix  $\mathcal{L} = \mathcal{A}_b \mathcal{P}^{-1} \mathcal{A}_b^T$  for  $\mathcal{G}$  can be written as

$$\mathcal{L} = \begin{bmatrix} \mathcal{A}_{1,b} \mathcal{P}_1^{-1} \mathcal{A}_{1,b}^T & 0 \\ 0 & \mathcal{A}_{2,b} \mathcal{P}_2^{-1} \mathcal{A}_{2,b}^T \end{bmatrix},$$

where  $\mathcal{P}_i^{-1}$  contains all the edge-covariance matrices belonging to the edges in  $\mathcal{G}_i$ . If one of the components, say  $\mathcal{G}_1$  does not have a reference node, then  $\mathcal{A}_{1,b} = \mathcal{A}_1$  and so  $\mathcal{A}_{1,b} \mathcal{P}_1^{-1} \mathcal{A}_{1,b}^T$  is singular, which makes  $\mathcal{L}$  singular. If both components have at least one reference node each, each of the diagonal blocks of  $\mathcal{L}$  is non-singular, which makes  $\mathcal{L}$  invertible. This proves the theorem. ■

*Proof of Lemma 2.2.2.* By definition, a linear estimate of the node variable  $x_u$  in the finite network  $(\mathcal{G}, P)$  is given by

$$\hat{x}_u = \sum_{e \in \mathcal{E}} j_e^T \zeta_e$$

for some matrices  $\{j_e, e \in \mathcal{E}\}$ . Therefore,

$$\hat{x}_u = \sum_{(p,q) \in \mathcal{E}} j_{p,q}^T (x_p - x_q + \epsilon_{p,q}) \tag{2.13}$$

which implies that

$$\begin{aligned}
\mathbb{E}[\hat{x}_u] &= \sum_{(p,q) \in \mathcal{E}} j_{p,q}^T (x_p - x_q) \\
&= \sum_{(p,q) \in \mathcal{E}} j_{p,q}^T x_p - \sum_{(p,q) \in \mathcal{E}} j_{p,q}^T x_q \\
&= \sum_{\bar{p} \in \mathcal{V}} \sum_{\substack{(p,q) \in \mathcal{E} \\ p=\bar{p}}} j_{p,q}^T x_p - \sum_{\bar{q} \in \mathcal{V}} \sum_{\substack{(p,q) \in \mathcal{E} \\ q=\bar{q}}} j_{p,q}^T x_q \\
&= \left( \sum_{\bar{p} \in \mathcal{V}} x_{\bar{p}}^T \sum_{\substack{(p,q) \in \mathcal{E} \\ p=\bar{p}}} j_{p,q} \right)^T - \left( \sum_{\bar{q} \in \mathcal{V}} x_{\bar{q}}^T \sum_{\substack{(p,q) \in \mathcal{E} \\ q=\bar{q}}} j_{p,q} \right)^T. \tag{2.14}
\end{aligned}$$

If  $j$  is a flow with intensity  $I_k$  from  $u$  to  $o$ , using (2.10) we conclude that the first term above can be expressed as

$$\begin{aligned}
\left( \sum_{\bar{p} \in \mathcal{V}} x_{\bar{p}}^T \sum_{\substack{(p,q) \in \mathcal{E} \\ p=\bar{p}}} j_{p,q} \right)^T &= (x_u - x_o)^T + \left( \sum_{\bar{p} \in \mathcal{V}} x_{\bar{p}}^T \sum_{\substack{(q,p) \in \mathcal{E} \\ p=\bar{p}}} j_{q,p} \right)^T \\
&= (x_u - x_o)^T + \left( \sum_{\bar{q} \in \mathcal{V}} x_{\bar{q}}^T \sum_{\substack{(p,q) \in \mathcal{E} \\ q=\bar{q}}} j_{p,q} \right)^T.
\end{aligned}$$

Combining this with (2.14), we get  $\mathbb{E}[\hat{x}_u] = (x_u - x_o) = x_u$ , because  $x_o = 0$ , which proves sufficiency.

If  $j$  is not a flow, there is at least one node, say  $r \in \mathcal{V}$ , where the flow condition (2.10) is violated. Assume for the moment that  $r$  is neither  $u$  nor  $o$ . We rewrite (2.13) as

$$\mathbb{E}[\hat{x}_u] = \sum_{\substack{(p,q) \in \mathcal{E} \\ p=r}} j_{p,q}^T (x_p - x_q) + \sum_{\substack{(p,q) \in \mathcal{E} \\ q=r}} j_{p,q}^T (x_p - x_q) + T$$

where  $T$  denotes the remaining terms of the sum and does not involve  $x_r$ ,

$$= \left( \sum_{\substack{(p,q) \in \mathcal{E} \\ p=r}} j_{p,q} - \sum_{\substack{(p,q) \in \mathcal{E} \\ q=r}} j_{p,q} \right)^T x_r + T_2,$$

where the terms constituting  $T_2$  also do not involve  $x_r$ . Since the flow condition (2.10) is not satisfied at  $r$ , the coefficient of  $x_r$  above is not zero and so  $\hat{x}_u$  is biased. The same proof technique can be applied to the case when  $r$  is either  $u$  or  $o$ , which proves necessity.

If  $j$  is an unbiased estimator of  $x_u$  in the finite network  $(\mathcal{G}, P)$ , the covariance of the estimation error is

$$\begin{aligned} \mathbb{E}[(x_u - \hat{x}_u)(x_u - \hat{x}_u)^T] &= \mathbb{E}\left[\left(\sum_{l=1}^{|\mathcal{E}|} j_l^T \epsilon_l\right)\left(\sum_{\ell=1}^{|\mathcal{E}|} j_\ell^T \epsilon_\ell\right)^T\right] \\ \Sigma_{u,o} &= \sum_{l=1}^{|\mathcal{E}|} j_l^T \mathbb{E}[\epsilon_l \epsilon_l^T] j_l = \sum_{l=1}^{|\mathcal{E}|} j_l^T P_l j_l, \end{aligned}$$

where the second inequality was obtained by using the fact that the measurement errors on different edges are uncorrelated. This proves the second statement of the lemma. ■

Now we are ready to prove Lemma 2.2.1.

*Proof of Lemma 2.2.1.* Without loss of generality, assume that there is only one reference node:  $\mathcal{V}_r = \{o\}$ . If  $\mathcal{G}$  is weakly connected, we can construct an undirected path  $\mathcal{P}$  from  $u$  to  $o$  and define an edge-function  $j$  as follows:

$$j_e^{\text{path}} = \begin{cases} \mathbf{j} & e \in \mathcal{P}, \vec{e} = \vec{\mathcal{P}} \\ -\mathbf{j} & e \in \mathcal{P}, \vec{e} \neq \vec{\mathcal{P}} \\ 0 & e \notin \mathcal{P} \end{cases}$$

where  $\vec{e} = \vec{\mathcal{P}}$  means that the *orientation* of the edge  $e$  is the same as the orientation of the path  $\mathcal{P}$ , and  $\vec{e} \neq \vec{\mathcal{P}}$  means the orientations are opposite. The orientation of an edge  $e$  in a path  $\mathcal{P} = \dots, p, e, q, \dots$  is said to be the same as the orientation of the path if  $e = (p, q)$ . If  $e = (q, p)$ , the orientation of the edge is opposite to

that of the path. It is straightforward to see that  $j$  is a flow of intensity  $I_k$  from  $u$  to  $o$ , and therefore by Lemma 2.2.2,  $j$  is an unbiased estimator of  $x_u$ .

If  $\mathcal{G}$  is not weakly connected, it can be decomposed into a number of disjoint subgraphs such that every one of them is weakly connected. These subgraphs are called weakly connected components of  $\mathcal{G}$ . Pick such a weakly connected component that does not contain the node  $o$ , call it  $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ , and pick an arbitrary node  $u$  in  $\mathcal{G}_1$ . By contradiction, assume that there exists a flow of matrix intensity  $I_k$  from  $u$  to  $o$ . Let  $\mathcal{A}_1$  be the generalized incidence matrix of  $\mathcal{G}_1$ . Let  $\mathcal{G}_1$  consist of  $N$  nodes and  $M$  edges, and without loss of generality, let  $u$  be numbered as node 1. Define  $\mathcal{J} := [j_1^T, j_2^T, \dots, j_M^T]^T \in \mathbb{R}^{kM \times k}$  as the tall matrix of the flows on the edges in the component  $\mathcal{G}_1$  and  $\boldsymbol{\omega} = [I_k, 0, \dots, 0]^T \in \mathbb{R}^{kN \times k}$  with  $I_k$  in the 1<sup>st</sup>  $k \times k$  block position, and 0 everywhere else. Then the conservation law (2.10) can now be expressed compactly as

$$\mathcal{A}_1 \mathcal{J} = \boldsymbol{\omega}. \quad (2.15)$$

Now define  $\mathbf{1} := [I_k, I_k, \dots, I_k]^T \in \mathbb{R}^{kN \times k}$ , and multiply both sides of the equation above by  $\mathbf{1}^T$ , which is equivalent to adding all the rows. It follows from Proposition 2.2.2 that the sum of each row in  $\mathcal{A}_1$  is 0. Therefore we obtain the following contradiction

$$\mathbf{1}^T \mathcal{A}_1 \mathcal{J} = \mathbf{1}^T \boldsymbol{\omega} \quad \Rightarrow \quad 0 = I_k. \quad (2.16)$$

Thus no flow of intensity  $I_k$  from  $u$  to  $o$  is possible. The result then follows from Lemma 2.2.2. ■

## Chapter 3

# Distributed algorithms for optimal estimation

In this chapter we answer the distributed algorithm question raised in Section 1.1, concerning computation of the optimal estimates of the node variables in a distributed way. We show that this objective is indeed feasible, and present two distributed asynchronous algorithms that achieve this goal. The algorithms are iterative, whereby every node starts with an arbitrary initial guess for its variable and successively improves its estimate by using two pieces of local information: the measurements available to it as well as the estimates of the nearby nodes. The latter can be obtained by communicating with the nearby nodes. The algorithms are guaranteed to converge to the optimal estimate when the number of iterations goes to infinity, as long as certain conditions of the inter-node communication are satisfied. The second algorithm is designed to have a faster convergence rate compared to the first. Both algorithms are robust to link failures, and they converge even in the presence of temporary faults of nodes and communication links.

Both algorithms require each node to have embedded communication and computation capability. As a result, these algorithms are not applicable to the case when some of the nodes in the measurement graph are not physical entities. An example of such an application is the mobile robot localization problem discussed in Section 2.1.4. Moreover, the measurement graph is assumed fixed in time. In some applications, such as the sensor network localization problem discussed in Section 2.1.1, this will mean the nodes are static. In other applications, a time-invariant measurement graph does not preclude motion of the nodes. For example, in both the time-synchronization and heading estimation problems discussed in Section 2.1.2 and Section 2.1.3, nodes can be moving without introducing any time variation in the measurement graph. *In this chapter we only consider measurement graphs that are time-invariant.*

*Organization:* Section 3.1 describes the constraints on computation and inter-node communication that an algorithm must abide by in order to qualify as a distributed algorithm. In Section 3.3 we describe the Jacobi algorithm and analyze its properties, including correctness, convergence rate, computation-communication trade-off and the effect of asymmetry in inter-node communication. Section 3.4 describes the OSE algorithm and analyzes its convergence properties. In section 3.5 we specifically discuss the effects of asymmetric communication. The chapter concludes with a discussion of open issues in Section 3.6.

## 3.1 Problem statement

Since the optimal estimate is the solution to the system of linear equations (2.6):  $\mathcal{L}\hat{\mathbf{x}} = \mathbf{b}$ , we seek iterative algorithms to compute its solution. We assume that every node  $u \in \mathcal{V}$  is a physical entity with a unique identifier that has the

capability to carry out computations and communicate information with a set of nearby nodes. We take the index of a node (nodes are indexed from 1 through  $n$ , where  $n$  is the number of nodes) as the unique identifier of the node. An algorithm qualifies as a distributed algorithm only if it satisfies the constraint that every node computes its own estimate and the information needed to carry out the computation is obtained by communication with its nearby nodes.

In order to describe the phrase “communication with nearby nodes” precisely, we define the *communication graph*  $\mathcal{G}^c = (\mathcal{V}, \mathcal{E}^c)$  associated with the measurement graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , which is a directed graph that consists of the same node set as the measurement graph but with a (typically) different edge set, whose edge directions determine which nodes can receive information from which other nodes. In particular, a node  $u$  can receive information from another node  $v$  if and only if there is an edge  $(v, u) \in \mathcal{E}^c$ . Note that when an edge  $(v, u)$  exists in the communication graph, the reverse edge  $(u, v)$  may not exist, in which case  $u$  can receive messages from  $v$  but not vice versa. Such asymmetry is quite common in wireless communication [63]. Asymmetry in communication could be caused, especially in ad-hoc wireless networks, due to inhomogeneous interference, packet collisions, and imperfect sleep scheduling arising from inaccurate time-synchronization. Communication between two nodes  $u$  and  $v$  is called *symmetric* if and only if both  $(u, v)$  and  $(v, u)$  belong to  $\mathcal{E}^c$ .

One has to keep in mind, though, that in certain situations, a relative measurement between a pair of nodes can be obtained only when the communication between them is symmetric. The relative clock offset measurement technique described in Section 2.1.2 (case A.) is one such example. However, there are situations when relative measurements can be obtained without symmetric communication. The relative positions obtained from range and bearing measure-

ments described in Section 2.1.1, and relative clock offset measurements by the RBS method of [1], described in Section 2.1.2, are such examples. Therefore it is important to allow the possibility of asymmetric communication in designing distributed algorithms.

Now we can describe precisely what we mean by a distributed algorithm. An iterative algorithm devised to compute the optimal estimate  $\hat{\mathbf{x}}^*$  is called *distributed* if it satisfies the following constraints:

- Constraint 3.1 (Distributed).*
1. Every node has knowledge of the relative measurements (and associated error covariances) corresponding to the edges in the measurement graph that are incident on itself.
  2. At every iteration, each node is allowed to receive a message from the nodes in its *1-hop in-neighborhood in the communication graph*,  $\mathcal{N}_u^c$ , which is defined as:

$$\mathcal{N}_u^c = \{v \in \mathcal{V}, (v, u) \in \mathcal{E}^c\}. \quad (3.1)$$

3. Each node is allowed to perform computations involving only variables that are local to the node or that were previously obtained.

The following assumptions are assumed to hold for every distributed algorithm considered in this chapter.

- Assumption 3.1.1.**
1. The measurement graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is weakly connected with respect to its reference nodes and does not contain parallel edges.
  2. The communication graph  $\mathcal{G}^c = (\mathcal{V}, \mathcal{E}^c)$  is such that for every pair of nodes that have a measurement edge between them, there is at least one communication edge between them.

3. If there is no measurement edge between a pair of nodes, then there is no communication edge between them, and no communication edge is directed toward a reference node.
4. Every node that is not a reference node has at least one communication edge directed toward it. □

The assumption of not having parallel edges is not restrictive because multiple measurements between the same pair of nodes can be combined into a single measurement (see Section 2.2.3). The second condition ensures that the nodes employing the algorithm will be able to use all the available measurements. The third condition clarifies that the communication graph is used only to model the information exchange that occurs during the execution of the algorithm. The fourth condition ensures that every node (other than a reference node) is able to receive messages from at least one neighbor, since otherwise it cannot update its estimate.

The communication graph  $\mathcal{G}^c$  is called *symmetric* if whenever  $(u, v) \in \mathcal{E}^c$ , where  $u, v$  are not reference nodes, we also have  $(v, u) \in \mathcal{E}^c$ . If there is at least one communication edge  $(u, v) \in \mathcal{E}^c$  such that  $(v, u) \notin \mathcal{E}^c$ , then the communication graph  $\mathcal{G}^c$  is called *asymmetric*. Recall that an edge  $e$  (in  $\mathcal{G}$  or  $\mathcal{G}^c$ ) between two nodes  $u$  and  $v$  is said to be incident on both the nodes  $u$  and  $v$ , which is denoted by  $e \sim u$  and  $e \sim v$ , respectively, whether the edge is directed from  $u$  to  $v$  or otherwise.

Figure 3.1 shows a measurement graph  $\mathcal{G}$  and an associated communication graph  $\mathcal{G}^c$ . The lack of communication from 2 to 1 does not introduce asymmetry since the reference node 1 does not use any information from its neighbors (see Assumption 3.1.1). According to our terminology the communication graph in

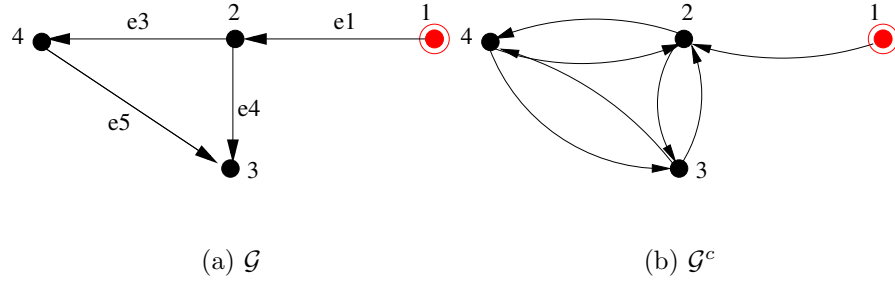


Figure 3.1. A measurement graph and a symmetric communication graph associated with it. Though there is no communication edge from 2 to 1, that is not a cause of asymmetry by Assumption 3.1.1.

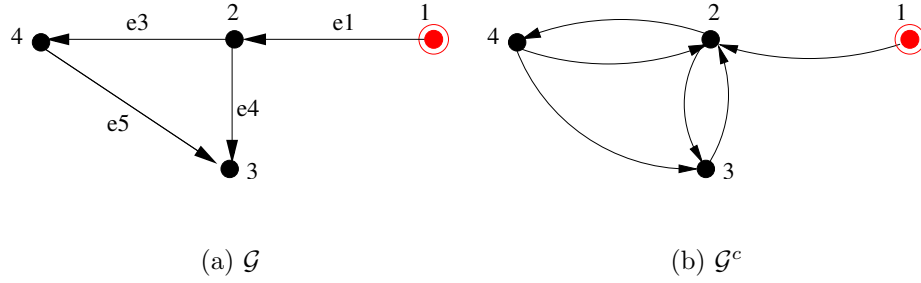


Figure 3.2. A measurement graph and an asymmetric communication graph associated with it. The asymmetry in the communication graph comes from the lack of a communication edge from node 3 to node 4.

the figure is therefore symmetric. Figure 3.2 shows another example of a measurement graph and a communication graph, where the communication graph is now asymmetric. The asymmetry comes from the lack of a communication edge  $(3, 4)$ , which means that node 3 can receive broadcasts from 4 but not the other way around.

Given an iterative algorithm, let  $\hat{\mathbf{x}}^{(i)} = [\hat{x}_2^{(i)T}, \dots, \hat{x}_{n_b}^{(i)T}]^T$ , where  $n_b$  is the number of non-reference nodes, be the vector of node estimates after the  $i$ th iteration of the algorithm has been completed. One iteration is said to be complete when

all nodes update their estimate once. The algorithms' *error at the  $i^{\text{th}}$  iteration* with respect to the BLU estimate  $\hat{\mathbf{x}}^*$  is

$$\mathbf{e}^{*(i)} := \hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*. \quad (3.2)$$

The algorithm is said to be *correct* if the error  $\mathbf{e}^{(i)} \rightarrow 0$  as  $i \rightarrow \infty$  for every initial condition  $\hat{\mathbf{x}}^{(0)}$ . We also define the *error ratio at the  $i^{\text{th}}$  iteration*

$$\varepsilon^{(i)} := \frac{\|\hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*\|}{\|\hat{\mathbf{x}}^{(0)} - \hat{\mathbf{x}}^*\|}. \quad (3.3)$$

The number of iterations  $i$  required so that the error ratio  $\varepsilon^{(i)}$  attains a value lower than  $\epsilon$ , denoted by  $n_{\text{iter}}(\epsilon)$ , is used as a measure of the convergence rate of the algorithm. The error ratio is particularly useful in comparing the convergence rate of two algorithms that start with the same initial estimates.

## 3.2 Contribution and prior work

In this chapter, we propose two distributed algorithms, namely, the Jacobi algorithm and the overlapping subgraph estimator (OSE) algorithm, to compute the BLU estimates of the node variables from the relative measurements. The algorithms are distributed in the sense that they satisfy Constraint 3.1. Parts of the this chapter's material have been reported in the papers [64–68]. The results of this chapter are summarized below.

1. We show that the Jacobi algorithm converge to the optimal estimate when the communication graph is symmetric. When the communication graph has asymmetry, then the algorithm converges to a sub-optimal estimate as long as certain conditions on the communication graph are satisfied. These

conditions have to do with the flow of information from the reference nodes to rest of the nodes.

The number of iterations required by the the Jacobi algorithm to make the error ratio lower than a specified value is established in terms of algebraic properties of the graph.

The Jacobi algorithms is proved to be robust to temporary node and communication failures and to asynchronous computation. That is, the estimates produced by the algorithm when time goes to infinity does not change even if the nodes communicate and update their estimates in an asynchronous manner, and some nodes and communication edges fail temporarily. A special structure of the measurement error covariance matrices were assumed (described in Assumption 3.3.1).

2. The OSE algorithm is proved to converge to the optimal estimate when the communication graph is symmetric. The algorithm is also shown to be robust to temporary communication-link and node failures, under the assumption of a special structure of covariance matrices.
3. We compare the energy consumption of three algorithms, Jacobi, EPA [69], and OSE, when nodes exchange information through wireless communication. Through simulations with a simple yet realistic model of energy consumption, we see that the OSE algorithm can drastically cut down the total energy expended to reach within a specified level of the optimal estimates.
4. We also show that in the presence of communication asymmetry, it is impossible to design a distributed algorithm satisfying the Constraint 3.1.

Convergence results for both the algorithms with asynchronous communication have been obtained under the assumption that the measurement error covariances have a special structure (described in Assumption 3.3.1). Simulations indicate that the algorithms converge even when the assumption is violated, but a proof is still lacking.

*Prior work:* Although the problem of localization has been extensively studied in the last 10 years, fueled by the explosive interest in sensor networks, most of the work on the topic has concentrated on estimating node locations from range measurements alone, and a few on estimating node locations from angle measurements alone [39, 43–50]. Estimating locations from both range and angle measurements, which is equivalent to localization from relative position measurements, have attracted attention only recently (see [52] and [51]). However, distributed algorithms to compute the BLUE location estimates have not investigated.

There is a rich literature on time-synchronization in a network of processors. The NTP(Network Time Protocol) is a forerunner in time synchronization protocols developed for the wired Internet, but is less desirable for wireless sensor networks due to its high energy consumption [70]. Many protocols have been developed in recent year for wireless networks, which include the RBS(Reference Broadcast System) [1], the TPSN(Timing-sync Protocol for Sensor Networks) [71], and the FTSP(The Flooding Time Synchronization Protocol) [56], to name a few. However, none of these synchronization protocols attempt to compute the optimal estimates of clock skews and offsets; rather they use a single path from a node to the reference to estimate those node variables. For example, in TPSN protocol [71], nodes close to a root node, called level 1 nodes, synchronize their clocks to the root node’s clock by using relative offset measurements. The nodes close to level 1 nodes, called level 2 nodes, in turn synchronize their clocks to the

level 1 nodes' clocks, and so on until all the nodes are synchronized. See [72] for a review of time synchronization protocols for sensor networks. To the best of our knowledge, Karp *et al.* [73] were the first to allude to a distributed algorithm for computing the optimal clock offset estimates. However, the algorithm was merely suggested, not analyzed in [73]. A distributed algorithm was proposed in [74] later for estimating time-offsets from all available relative offset measurements.

The EPA algorithm proposed by Delouille *et al.* [69], though for a completely different application, can also be used to compute the BLU estimates in a distributed manner. We compare the Jacobi and OSE algorithms to the EPA algorithm through simulations in Sections 3.3.3 and 3.4.1.

Since the optimal estimate is a solution of a system of linear equations (2.6), we exploit iterative techniques of solving linear equations, which has a long and rich history [75]. In fact, the algorithms proposed by both Karp *et al.* [73] and Giridhar and Kumar [74] for distributed time synchronization are based on the Jacobi method of iteratively solving linear equations [75]. Our first algorithm for computing the optimal estimates from relative measurements is also based on the Jacobi method. In that respect, the Jacobi algorithm proposed in this dissertation is not novel. However, we provide a thorough analysis of the effect of asymmetric communication on the Jacobi algorithm's convergence properties. In contrast, in the algorithm proposed by Giridhar and Kumar [74] for estimating time-offsets, the effect of such asymmetry was overlooked. The EPA algorithm proposed in [69] is a block-Jacobi method [75] of solving linear equations.

Apart from the Jacobi method, there are many other iterative methods of solving linear equations, some of them having a faster convergence rate, such as Gauss – Siedel, SOR, and conjugate gradient methods [75]. However, not all of these methods are applicable in devising distributed algorithms, since the information

required by these algorithms to carry out the computations may demand an unacceptable level of communication between nodes or the interference by a central authority. For example, the Gauss-Siedel method requires that the variables be updated in a specific sequence. In a network of devices that exchange information with one another through wireless communication, ensuring such an order while satisfying the Constraint 3.1 may be quite difficult. However, the Weighted Additive Schwarz method [76] offers potential for distributed implementation. The OSE algorithm described in this dissertation is closely related to the multisplitting and Weighted Additive Schwarz method of solving linear equations [76].

### 3.3 Jacobi algorithm

In the Jacobi algorithm, a node obtains multiple estimates of its own variable by adding the appropriate relative measurements to its neighbors' estimates. It then computes the new estimate of its variable by taking a weighted average of those estimates. To describe the algorithm, we first define the *1-hop measurement neighborhood of  $u$* , denoted by  $\mathcal{N}_u$ , as the set of nodes with which  $u$  shares a measurement edge (irrespective of the direction):

$$\mathcal{N}_u = \{v \in \mathcal{V} | (u, v) \in \mathcal{E} \text{ or } (v, u) \in \mathcal{E}\} = \{v \in \mathcal{V} | d_{\mathcal{G}}(u, v) = 1\}, \quad (3.4)$$

where  $d_{\mathcal{G}}$  is the *graphical distance* between the nodes  $u$  and  $v$ , which is the number of edges that have to be traversed in going from one to the other. The graphical distance is evaluated without regards to the edge directions. The *Jacobi algorithm* for computing the optimal estimates of the node variables is an iterative algorithm that operates as follows for each node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ .

1. node  $u$  picks an arbitrary initial estimate  $\hat{x}_v^{(0)}$  for the node variables  $x_v$ ,  $v \in$

$\{u\} \cup \mathcal{N}_u^c$ , i.e., for itself and those of its neighbors from which it can receive messages. The neighbor set  $\mathcal{N}_u^c$  in the communication graph was defined in Constraint 3.1. These estimates need not be consistent across different nodes. The reference nodes start at their known values.

2. at the  $i^{\text{th}}$  iteration, node  $u$  assumes that the current estimate  $\hat{x}_v^{(i)}$  for the node variable  $x_v$  of each communication neighbor  $v \in \mathcal{N}_u^c$  is correct and updates its own estimate by solving the following equation:

$$\left( \sum_{e \in \vec{\mathcal{E}}_u(1)} P_e^{-1} \right) \hat{x}_u^{(i+1)} = \sum_{e \in \vec{\mathcal{E}}_u(1)} P_e^{-1} \left( \hat{x}_{e \setminus u}^{(i)} + a_{u,e} \zeta_e \right), \quad (3.5)$$

where  $e \setminus u$ , for an edge  $e$  incident on  $u$ , denotes the “other end” of  $e$ <sup>1</sup>, and  $a_{ue}$  is the  $(u, e)^{\text{th}}$  entry of the incidence matrix  $A$  (defined in Section 2.2.1), and  $\vec{\mathcal{E}}_u(1)$  is the set of edges in the measurement graph  $\mathcal{G}$  that are incident on the node  $u$  such that there are communication edges from their other ends toward  $u$  :

$$\vec{\mathcal{E}}_u(1) := \{e \in \mathcal{E} \mid e \sim u, (e \setminus u, u) \in \mathcal{E}^c\}. \quad (3.6)$$

The arrow in the notation is used to emphasize that the measurement edges in  $\vec{\mathcal{E}}_u(1)$  depend on the direction of the edges in the communication graph. After the computation, node  $u$  broadcasts the new estimate  $\hat{x}_u^{(i+1)}$  to all its neighbors that can receive messages from it.

3. At the end of the  $i^{\text{th}}$  iteration, node  $u$  listens for the broadcasts from its 1-hop in-neighbors, which are used to update the node variable estimate  $\hat{x}_v^{(i+1)}$  for each  $v \in \mathcal{N}_u^c$ . Once all updates are received, a new iteration can start.

---

<sup>1</sup>That is, if  $e = (v, u)$ , then  $e \setminus u = v$  and  $e \setminus v = u$ .

These iterations can be terminated at a node when the change in its recent estimate is seen to be lower than a pre-specified threshold value or a pre-specified maximum number of iterations are completed. Figure 3.3 shows the relevant equations for one iteration of the Jacobi algorithms applied to the measurement graph shown in Figure 2.6.

To gain insight into the Jacobi algorithm, imagine for the moment that when node  $u$  receives from its communication neighbors their current estimates  $\hat{x}_v^{(i)}$ ,  $v \in \mathcal{N}_u^c$ , it believes those are the optimal estimates of the corresponding node variables. In that case, node  $u$  can compute its optimal estimate by using the measurements between itself and its communication neighbors. This estimation problem is no different from the original BLUE estimation problem, except that it is defined over the much smaller graph  $\vec{\mathcal{G}}_u(1) = (\mathcal{V}_u^c(1), \vec{\mathcal{E}}_u(1))$ , whose nodes include  $u$  and its communication neighbors:

$$\mathcal{V}_u^c(1) = \{u\} \cup \mathcal{N}_u^c(1).$$

We call  $\vec{\mathcal{G}}_u(1)$  the *1-hop communication-enabled subgraph of  $\mathcal{G}$  centered at  $u$* . Since  $u$  thinks that the node variables of its neighbors are exactly known, all of these nodes should be understood as references; so that  $\vec{\mathcal{G}}_u(1)$  has only one unknown node variable, namely,  $x_u$ . Node  $u$  can now compute an estimate of its node variable by solving the BLU estimation problem associated with the 1-hop subgraph  $\vec{\mathcal{G}}_u(1)$ , which turn out to be (3.5). The Jacobi algorithm can therefore be thought of as an algorithm in which every node solves a local optimal estimation problem by assuming their neighbors' estimates are correct and performing this computation repeatedly as those estimates are updated. The name “Jacobi algorithm” comes from the fact that when communication is symmetric, the update equation (3.5) is essentially the Jacobi method for solving the linear equation (2.6) that defines the optimal estimate [75].

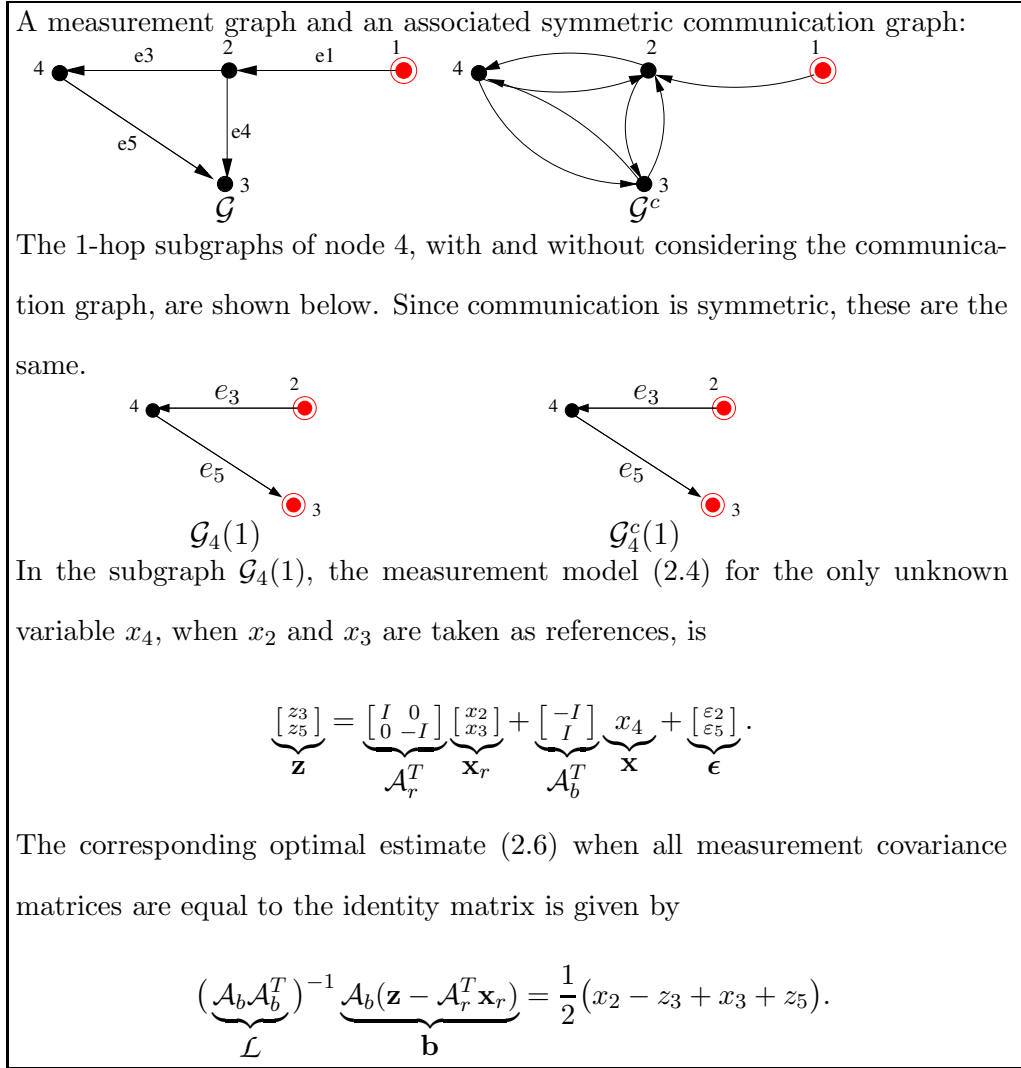


Figure 3.3. The Jacobi Iteration with symmetric communication. The iterations of the Jacobi algorithm are explained according to the interpretation that every node is repeatedly solving a local optimal estimation problem that is defined over the 1-hop subgraph centered at itself.

(continued from Figure 3.3) Since node 4 can receive messages from all of its measurement neighbors, the Jacobi iteration for node 4 is

$$\hat{x}_4^{(i+1)} = \frac{1}{2}(\hat{x}_2^{(i)} - z_3 + \hat{x}_3^{(i)} + z_5)$$

A similar construction based on the 1-hop subgraphs centered at nodes 2 and 3 leads to update equations for estimates of  $x_2$  and  $x_3$  given by

$$\begin{aligned}\hat{x}_2^{(i+1)} &= \frac{1}{4}(\hat{x}_4^{(i)} + \hat{x}_3^{(i)} + \zeta_3 + \zeta_4 - \zeta_1 - \zeta_2), \\ \hat{x}_3^{(i+1)} &= \frac{1}{2}(\hat{x}_2^{(i)} + \hat{x}_4^{(i)} - \zeta_4 - \zeta_5).\end{aligned}$$

The reference node, which is node 1, is assumed to be at the origin, and thus  $x_1$  does not appear in the equations.

Figure 3.4. Figure 3.3 contd.

### 3.3.0.1 Asynchronous implementation

The algorithm described by (3.15) is synchronous, since that description implicitly assumes that the iteration counter  $i$  is common to all the nodes. This means that all nodes update their estimates at the same time after getting updates from all of their communication neighbors. Moreover, the description above implicitly assumes that there are no communication faults, i.e., a node is always able to receive data from all of its communication neighbors. In practice, nodes may have varying processor power, so that one may be ready to start the next iteration while other nodes have not finished their computation. In addition, in wireless networks with time dependent communication failures or nodes with scheduled sleep-wake cycles, the communication graph may be time varying. In both these cases, waiting to get information from all neighbors may not be advisable. In this case, the algorithm can be implemented in an *asynchronous* fashion, in which nodes wait for a “time-out” period to receive estimates from their neighbors. If estimates from some neighbors do not arrive in this time, they use the previously received data from those nodes. Asynchronous implementation therefore requires a local buffer to store data that is successfully received from neighboring nodes for use later when communication fails.

Consider time index  $t \in \mathbb{N}$  that is incremented by 1 at the end of every time-out period. A communication edge  $(u, v)$  is said to fail in time  $t$  if during the time between  $t - 1$  and  $t$ , broadcasts from node  $u$  fail to reach  $v$ . A node  $u$  is said to fail at time  $t$  if during the time between  $t - 1$  and  $t$ , node  $u$  either does not broadcast its current estimate or does not process any information. Such node failures can occur due to sleep-scheduling [54], among other reasons. We do not consider other forms of node failure that is possible, such that a failed node sends incorrect or

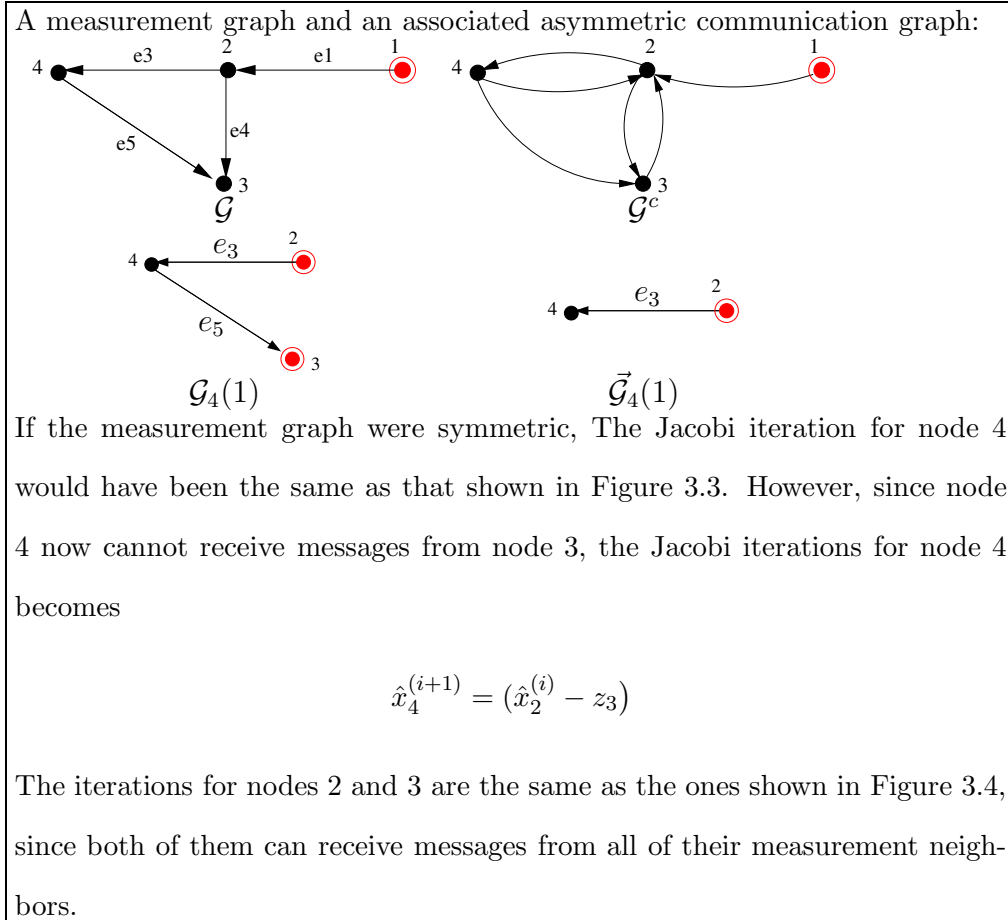


Figure 3.5. The Jacobi Iteration with asymmetric communication.

random data. We say that a communication edge  $(u, v)$  is *active* at time  $t$  if in that time, neither of the nodes  $u$  and  $v$  fails and the communication edge  $(u, v)$  does not fail. At every time  $t$ , the communication graph  $\mathcal{G}^c(t) = (\mathcal{V}, \mathcal{E}^c(t))$  consists of all the nodes of  $\mathcal{G}$  and all the communication edges that are active in that time.

Although buffers can partially mitigate the effects of time variation in the communication, certain difficulties still remain. Since every node  $u$  has to keep and update estimates of the variables of its 1-hop communication-neighbors (see (3.5)), node  $u$  has to know its communication-neighbors in advance. Consider the following situation: nodes  $u$  and  $v$  are neighbors in the measurement graph, but  $u$  does not receive any communication from  $v$  for the first, say, 10, iterations, and then receives a message from  $v$  at the 11<sup>th</sup> iteration, and never receives any communication from  $v$  thereafter. Should node  $u$  initialize another local variable for  $x_v$ 's current estimate at the 11<sup>th</sup> iteration? If so, what should it do when it sees that all communication from  $v$  has ceased thereafter? What if another neighbor appears at the 100<sup>th</sup> iteration?

To avoid such difficulties, we assume that every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$  detects its communication-neighbors during an initial detection phase, before the iterations begin. This detection, which may be carried out even during the process of obtaining the relative measurements, leads to an *initial communication graph*  $\mathcal{G}_{\text{init}}^c = (\mathcal{V}, \mathcal{E}_{\text{init}}^c)$  consisting of those communication edges that were active over the time interval of the detection phase. A node does not update its communication neighborhood  $\mathcal{N}_u^c$  thereafter.

### 3.3.1 Correctness and performance analysis of the Jacobi algorithm

To analyze the algorithm, we now define a few matrices. The *combined incidence matrix*  $A^c \in \mathbb{R}^{n \times m}$ , where  $m$  is the number of edges in the measurement graph  $\mathcal{G}$ , for the pair of directed graphs  $(\mathcal{G}, \mathcal{G}^c)$  is defined in the following manner:

$$[A^c]_{u,e} = \begin{cases} a_{u,e} & \text{if } e \in \mathcal{E}, e \sim u, (e \setminus u, u) \in \mathcal{E}^c \\ 0 & \text{otherwise} \end{cases}, \quad (3.7)$$

where  $a_{u,e}$  is the  $(u, e)^{\text{th}}$  entry of the incidence matrix  $A$  for the measurement graph  $\mathcal{G}$ . Recall that for an edge  $e$  that is incident on a node  $u$ ,  $e \setminus u$  denotes the other end of  $u$  (see Section 3.3). The incidence matrix was defined in Chapter 2 and is standard in algebraic graph theory. The combined incidence matrix  $A^c$  is not standard in graph theory, though. Note that  $A^c(\mathcal{G}, \mathcal{G}^c) = A(\mathcal{G})$  if and only if the communication graph  $\mathcal{G}^c$  is symmetric.

The *weighted in-degree matrix*  $\mathcal{D} \in \mathbb{R}^{kn \times kn}$  of the directed graph pair  $(\mathcal{G}, \mathcal{G}^c)$  is defined as a block-diagonal matrix with

$$[\mathcal{D}]_{u,u} = \sum_{e \in \mathcal{E}_u^c(1)} P_e^{-1} \quad (3.8)$$

Similarly, the *weighted adjacency matrix*  $\mathcal{C} \in \mathbb{R}^{kn \times kn}$  of the directed graph pair  $(\mathcal{G}, \mathcal{G}^c)$  is defined as

$$[\mathcal{C}]_{u,v} = \begin{cases} P_e^{-1} & \text{if } \exists e \in \mathcal{E} \text{ such that } e \sim u, e \sim v \text{ and } (v, u) \in \mathcal{E}^c, \\ 0 & \text{otherwise} \end{cases} \quad (3.9)$$

Let  $\mathcal{M}, \mathcal{N} \in \mathbb{R}^{kn_b \times kn_b}$  be the sub-matrices of  $\mathcal{D}$  and  $\mathcal{C}$ , respectively, obtained by removing the rows and columns corresponding to the reference nodes, where

$n_b = |\mathcal{V} \setminus \mathcal{V}_r|$  is the number of nodes that do not know their variables:

$$[\mathcal{M}]_{u,v} = \begin{cases} [\mathcal{D}]_{u,u} & \text{if } u = v, u \in \mathcal{V} \setminus \mathcal{V}_r, \\ 0 & u \neq v \end{cases}, \quad (3.10)$$

$$[\mathcal{N}]_{u,v} = [\mathcal{C}]_{u,v} \text{ for } u, v \in \mathcal{V} \setminus \mathcal{V}_r. \quad (3.11)$$

Figure 3.6 shows an example of these matrices. Now define

$$\mathcal{L}_c := \mathcal{D} - \mathcal{C}, \quad (3.12)$$

$$\mathcal{L}_c := \mathcal{M} - \mathcal{N}. \quad (3.13)$$

It is straightforward to verify that

$$\mathcal{L}_c = \mathcal{A}_b^c \mathcal{P}^{-1} \mathcal{A}_b^T. \quad (3.14)$$

where  $\mathcal{A}_b^c := A_b^c \otimes I_k$  and  $A_b^c$  is the *basis combined incidence matrix* of  $(\mathcal{G}, \mathcal{G}^c)$  obtained from  $\mathcal{A}^c$  by removing from it the rows that correspond to the reference nodes. It can be verified that  $\mathcal{L}_c$  does not depend on the edge directions of  $\mathcal{G}$  but does depend on that in  $\mathcal{G}^c$ . Figure 3.6 shows an example of all of the matrices described above.

We conclude the definitions with the following observation:

**Proposition 3.3.1.** *The matrix  $\mathcal{M}$  defined in (3.10) for a graph pair  $(\mathcal{G}, \mathcal{G}^c)$  is positive definite.*  $\square$

*Proof of Proposition 3.3.1.* Assumption 3.1.1 ensures that for every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is at least measurement edge  $e \in \mathcal{E}, e \sim u$  such that there is an incoming communication edge in  $\mathcal{G}^c$  from the other end of  $u$ , i.e.,  $\exists(e \setminus u, u) \in \mathcal{E}^c$ . As a result, the set  $\mathcal{E}_u^c(1)$  is non-empty for every  $u \in \mathcal{V} \setminus \mathcal{V}_r$ . The result now follows from the definition of  $\mathcal{D}$  and  $\mathcal{M}$  in (3.8) and (3.10).  $\blacksquare$

We start with the synchronous, time-invariant case, in which every node updates its estimate at the same time instant, the communication graph is fixed for all time (i.e., there are no node or edge failures). In this case, the Jacobi algorithm (3.5) can be compactly expressed as the following discrete-time dynamical system:

$$\begin{aligned}\mathcal{M}\hat{\mathbf{x}}^{i+1} &= \mathcal{N}\hat{\mathbf{x}}^i + \mathbf{b}_c, \\ \Rightarrow \hat{\mathbf{x}}^{(i+1)} &= \mathcal{M}^{-1}\mathcal{N}\hat{\mathbf{x}}^{(i)} + \mathcal{M}^{-1}\mathbf{b}_c,\end{aligned}\tag{3.15}$$

where

$$\mathbf{b}_c := \mathcal{A}_b^c \mathcal{P}^{-1}(\mathbf{z} - \mathcal{A}_r^T \mathbf{x}_r).\tag{3.16}$$

The fixed point of the iteration (3.15) is given by the solution of the following system of linear equations, when it exists:

$$\mathcal{L}_c \hat{\mathbf{x}}^\infty = \mathbf{b}_c,\tag{3.17}$$

For  $\hat{\mathbf{x}}^\infty$  to exist,  $\mathcal{L}_c$  must be invertible. In the next section we describe the conditions under which it is so, and when the Jacobi algorithm converges to this solution. It is important to note here that  $\mathcal{L}_c = \mathcal{L}$  if and only if all the inter-node communication is symmetric. If there is asymmetry in the communication graph, the fixed point of the Jacobi iteration will *not* coincide with the BLU estimates. This will be stated more precisely in the next section.

We end this section with the observation that when the limiting estimate  $\hat{\mathbf{x}}^\infty$  exists, it is an unbiased estimate of the node variable vector  $\mathbf{x}$ .

**Lemma 3.3.1.** *The unique solution to  $\mathcal{L}_c \hat{\mathbf{x}}^\infty = \mathbf{b}_c$ , when it exists, is an unbiased estimate of  $\mathbf{x}$  and the covariance of the estimation error  $\tilde{\mathbf{e}}^\infty := \mathbf{x} - \hat{\mathbf{x}}^\infty$  is given by*

$$\Sigma := \mathbb{E}[\tilde{\mathbf{e}}^\infty \tilde{\mathbf{e}}^{\infty T}] = \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} \mathcal{A}_b^{cT} \mathcal{L}_c^{-T}. \quad \square$$

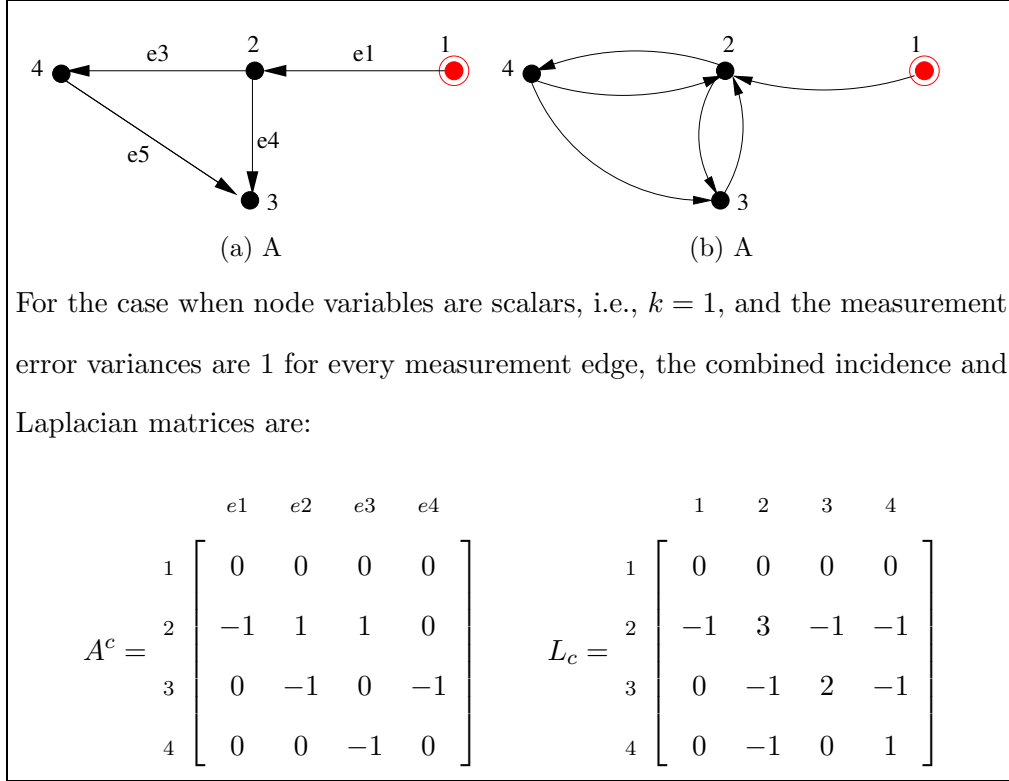


Figure 3.6. The matrices used to compactly represent the Jacobi algorithm with asymmetric communication, for the pair  $(\mathcal{G}, \mathcal{G}^c)$  shown. For simplicity, we considered the case  $k = 1$  and every measurement error variance is 1. We use  $M, N, D, C$  and  $L_c$  to represent  $\mathcal{M}, \mathcal{N}, \mathcal{D}, \mathcal{C}$  and  $\mathcal{L}_c$ , respectively, to emphasize that  $k = 1$ .

(continued from Figure 3.6) The in-degree and adjacency matrices of  $(\mathcal{G}, \mathcal{G}^c)$ :

$$C = \begin{array}{c} \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \end{array} & D = \begin{array}{c} \begin{array}{ccccc} & 1 & 2 & 3 & 4 \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \end{array}$$

and their submatrices obtained by removing the rows and columns corresponding to the reference nodes:

$$N = \begin{array}{c} \begin{array}{ccc} & 2 & 3 & 4 \\ \begin{array}{c} 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \end{array} & M = \begin{array}{c} \begin{array}{ccc} & 2 & 3 & 4 \\ \begin{array}{c} 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \end{array}$$

Figure 3.7. Figure 3.6 contd.

*Proof of Lemma 3.3.1.* From (3.17), (3.16), (3.14) and (2.4), we get

$$\begin{aligned}\hat{\mathbf{x}}^\infty &= \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} (\mathbf{z} - \mathcal{A}_r^T \mathbf{x}_r) \\ &= \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} (\mathcal{A}_b^T \mathbf{x} + \boldsymbol{\epsilon}) \\ &= \mathbf{x} + \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} \boldsymbol{\epsilon},\end{aligned}$$

where the last equality follows from (3.14). It follows that  $\mathbb{E}[\hat{\mathbf{x}}^\infty] = \mathbf{x}$ , so the estimate is unbiased, and also that  $\tilde{\mathbf{e}}^\infty = \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} \boldsymbol{\epsilon}$ . Therefore, the covariance of the estimation error  $\tilde{\mathbf{e}}^\infty$  is

$$\begin{aligned}\mathbb{E}[\tilde{\mathbf{e}}^\infty \tilde{\mathbf{e}}^{\infty T}] &= \mathbb{E}[\mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} \boldsymbol{\epsilon} \boldsymbol{\epsilon}^T \mathcal{P}^{-1} \mathcal{A}_b^{cT} \mathcal{L}_c^{-T}] \\ &= \mathcal{L}_c^{-1} \mathcal{A}_b^c \mathcal{P}^{-1} \mathcal{A}_b^{cT} \mathcal{L}_c^{-T},\end{aligned}$$

which proves the result. ■

### 3.3.1.1 Correctness of the Jacobi algorithm

First we analyze the simplest case of symmetric, time-invariant communication.

**Theorem 3.3.1.** *Consider the Jacobi algorithm implemented on a measurement and time-invariant communication graph pair  $(\mathcal{G}, \mathcal{G}^c)$  that satisfies Assumption 3.1.1. If the communication graph  $\mathcal{G}^c$  is symmetric, then the synchronous Jacobi algorithm is correct.* □

To prove Theorem 3.3.1, we will also need the following technical result from [77].

**Proposition 3.3.2 (Lemma 4.2 of [77]).** *Let  $X := D - N$  be a square matrix such that  $D + D^* > 0$  and  $X_\theta = D + D^* - (e^{i\theta} N + e^{-i\theta} N^*) > 0$  for all  $\theta \in \mathbb{R}$ . Then  $\rho(D^{-1} N) < 1$ , where  $\rho(\cdot)$  denotes the spectral radius.* □

*Proof of Theorem 3.3.1.* Since the communication graph is symmetric,  $\mathcal{L}_c = \mathcal{L}$  and  $\mathbf{b}_c = \mathbf{b}$ . Therefore  $\mathcal{L} = \mathcal{M} - \mathcal{N}$ , so that we can rewrite (2.6), the equation defining the BLU estimate, as

$$\begin{aligned}\mathcal{M}\hat{\mathbf{x}}^* &= \mathcal{N}\hat{\mathbf{x}}^* + \mathbf{b} \\ \Rightarrow \hat{\mathbf{x}}^* &= \mathcal{M}^{-1}\mathcal{N}\hat{\mathbf{x}}^* + \mathcal{M}^{-1}\mathbf{b},\end{aligned}$$

Comparing the above with (3.15), we see that the algorithm's error (defined in (3.2)) evolves according to

$$\mathbf{e}^{*(i+1)} = \mathcal{J}\mathbf{e}^{*(i)}, \quad (3.18)$$

where

$$\mathcal{J} := \mathcal{M}^{-1}\mathcal{N} \quad (3.19)$$

is the Jacobi iteration matrix. Therefore, to prove the theorem, we need to show that  $\rho(\mathcal{M}^{-1}\mathcal{N}) < 1$ . Since  $\mathcal{L} = \mathcal{M} - \mathcal{N} > 0$  and  $\mathcal{M} > 0$ , it follows that  $\mathcal{M} > \mathcal{N} \geq \cos \theta \mathcal{N}$  for every  $\theta \in \mathbb{R}$ . From Proposition 3.3.2 it follows that  $\rho(\mathcal{M}^{-1}\mathcal{N}) < 1$ , which proves the theorem.  $\blacksquare$

To analyze the correctness and performance of the algorithm when communication graph is asymmetric and possibly time-varying, we make an additional assumption:

**Assumption 3.3.1 (Diagonal).** Either of the following two conditions hold:

1. the measurement error covariance matrices are either all equal to one another, i.e.,  $\exists P_o \in \mathbb{S}^{k+}$  such that  $P_e = P_o, \forall e \in \mathcal{E}$ , or,
2. every measurement error covariance matrix is diagonal (but not necessarily equal to one another).  $\square$

The case of vector-valued variables can be reduced to the scalar valued case as long Diagonal Assumption 3.3.1 is satisfied, as we now show.

If the covariance matrices are all equal, then the covariances play no role and can be taken as the identity matrix. To see why, let  $P_e = P_o \in \mathbb{R}^{k \times k} \quad \forall e \in \mathcal{E}$ , then  $P = I_m \otimes P_o$ , where  $\otimes$  denotes the Kronecker product and  $m$  is the number of edges in  $\mathcal{G}$ . Equation (2.6) now simplifies to

$$L \otimes P_o^{-1} \hat{\mathbf{x}}^* = (A_b \otimes P_o^{-1}) \bar{\mathbf{z}},$$

where  $L := A_b A_b^T$  is the unweighted Dirichlet (grounded) Laplacian matrix of the graph  $\mathcal{G}$ . Simple algebraic manipulation using rules of Kronecker algebra show that the solution to this equation is  $\hat{\mathbf{x}}^* = ((L_c^{-1} A_b) \otimes I_k) \bar{\mathbf{z}}$ . Therefore we need only to solve  $(L_b \otimes I_k) \hat{\mathbf{x}}^* = (A_b \otimes I_k) \bar{\mathbf{z}}$  to get the optimal estimate. We can decompose the above following  $k$  systems of *decoupled* equations:

$$L_b \hat{\mathbf{x}}_j^* = \mathbf{b}_j, \quad j = 1, \dots, k, \quad (3.20)$$

where  $\mathbf{x}_j := [x_{1,j}, \dots, x_{n_b,j}]^T \in \mathbb{R}^{n_b}$  is the vector of the  $j^{\text{th}}$  component of the node variables,  $\mathbf{b}_j = A_b \bar{\mathbf{z}}_j$  with  $\bar{\mathbf{z}}_j := [\bar{\zeta}_{1,j}, \dots, \bar{\zeta}_{m,j}]^T \in \mathbb{R}^m$  being the vector of the  $j^{\text{th}}$  components of the entries in  $\bar{\mathbf{z}}$ . When the covariance matrices are diagonal, it is again straightforward to show that the estimation problem for each of the  $k$  components of every node variable is decoupled. In light of the discussion above, *when the Diagonal Assumption 3.3.1 holds, we only need to consider the case when the variables and measurements are scalar-valued.*

In the sequel, to emphasize that we are proving results only for the scalar valued variables, we denote the matrices  $\mathcal{L}_c, \mathcal{M}$ , and  $\mathcal{N}$  in the special case  $k = 1$  by  $L_c, M$ , and  $N$ . One of the main advantages of considering the scalar case of  $k = 1$  is that  $L_c$  turns out to be an M-matrix [78, Chapter 6]. M-matrices are a

special class of matrices that have been widely studied in the context of iterative methods of solving linear equations since they arise naturally in numerical solution of PDEs, in convergence analysis of matrix-iteration processes, and they possess a number of useful properties that help establish convergence of asynchronous parallel iterative methods [78, 79]. Non-singular M-matrices are termed *matrices of class K* by Fiedler and Ptak [80].

To define M-matrices, we start with matrices of class Z, which are square matrices with non-positive off-diagonal entries. A matrix  $X$  of class Z is an M-matrix if it can be written as  $X = sI - B$  where  $B \succeq 0$  and  $s \geq \rho(B)$ . Here and in the sequel,  $\succ (\succeq)$  is used to denote entry-wise ordering. That is, for a matrix or a vector  $X$ ,  $X \succ (\succeq) 0$  means every entry of  $X$  is positive (non-negative). For more information on M-matrices, the reader is referred to Chapter 6 of [78], where 50 equivalent characterizations of M-matrices are provided. We start with the following technical result:

**Lemma 3.3.2.** *Consider the matrices  $L_c, M, N \in \mathbb{R}^{n_b \times n_b}$  (i.e., the matrices  $\mathcal{L}_c, \mathcal{M}, \mathcal{N}$  for the case  $k = 1$ ) defined in (3.12), (3.10), and (3.11) for a measurement and time-invariant communication graph pair  $(\mathcal{G}, \mathcal{G}^c)$  that satisfies Assumption 3.1.1. The matrix  $L_c$  defined for  $(\mathcal{G}, \mathcal{G}^c)$  is an M-matrix. Moreover, the following statements are equivalent:*

1. *The matrix  $L_c$  is a non-singular M-matrix.*
2.  *$\rho(J) < 1$ , where  $J := M^{-1}N$  is the Jacobi iteration matrix (for  $k = 1$ ) and  $\rho(\cdot)$  denotes the spectral radius.*
3. *For every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is a directed path in the communication graph  $\mathcal{G}^c$  from at least one of the reference nodes to  $u$ .* □

A *directed path*  $\mathcal{P}$  from a node  $p_1$  to another node  $p_m$  in a graph  $\mathcal{G}$  is an alternating sequence of finite number of nodes and edges that start with  $p_1$  and end with  $p_m$ :

$$\mathcal{P} = \{p_1, e_1, p_2, e_2, \dots, p_l, e_l, p_{l+1}, \dots, p_{m-1}, p_m\}$$

such that every edge  $e_l$  in the path is directed from the previous node to the next node in the path:  $e_l = (p_l, p_{l+1})$ . Note that edge directions matter in this definition.

The proofs of lemma 3.3.2 and the theorem below, which describes the behavior of the Jacobi algorithm with (possibly) asymmetric communication, are provided in Section 3.7.

**Theorem 3.3.2.** *Consider the synchronous Jacobi algorithm implemented on a measurement graph  $\mathcal{G}$  and its associated time-invariant communication graph  $\mathcal{G}^c$  such that  $\mathcal{G}$  and  $\mathcal{G}^c$  satisfy Assumptions 3.1.1. Furthermore, assume that Diagonal Assumption 3.3.1 is satisfied. The Jacobi algorithm converges to the unique solution of  $\mathcal{L}_c \hat{\mathbf{x}}^\infty = \mathbf{b}_c$ , if and only if, for every  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is a directed path in  $\mathcal{G}^c$  from at least one of the reference nodes to  $u$ .  $\square$*

*Proof of Theorem 3.3.2.* We only consider the case  $k = 1$  in the proof. The case for  $k > 1$  will follow from Diagonal Assumption 3.3.1, as explained earlier.

Define the error at the  $i^{\text{th}}$  iteration as the difference between the current estimate and the limiting estimate:

$$\mathbf{e}^{(i)} := \hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^\infty.$$

We rewrite (3.17) as

$$M\hat{\mathbf{x}}^\infty = N\hat{\mathbf{x}}^\infty + \mathbf{b}_c$$

It follows from (3.15) that the error evolves according to :

$$\mathbf{e}^{(i+1)} = M^{-1}N\mathbf{e}^{(i)} \quad (3.21)$$

Clearly, the Jacobi algorithm converges (i.e., the error  $\mathbf{e}^{(i)} \rightarrow 0$  as  $i \rightarrow \infty$ ) if and only if  $\rho(M^{-1}N) < 1$ . It follows from Lemma 3.3.2 that this condition is satisfied if and only if, for every  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is a directed path in  $\mathcal{G}^c$  from at least one reference nodes to  $u$ , which proves the theorem.  $\blacksquare$

The next theorem states how the algorithm behaves in the asynchronous mode, i.e., when there are temporary node and communication edge failures, and nodes update their estimates in an asynchronous manner as described in Section 3.3.0.1. Apart from deterministic failures, we consider the following model of random node and communication edge failures. At every time instant  $t \in \mathbb{N}$ , every communication edge can fail independently of all other links with probability  $p$ , and every node can fail independently of all other nodes, with probability  $q$ , where  $p < 1, q < 1$ . This model of failure is referred to as *i.i.d. failure*. The proof the result is provided in Section 3.7.

**Theorem 3.3.3.** *Consider the asynchronous Jacobi algorithm implemented on the measurement graph  $\mathcal{G}$  and its associated time-varying communication graph  $\mathcal{G}^c(t)$ , such that  $\mathcal{G}$  and  $\mathcal{G}^c(t)$  satisfy Assumption 3.1.1 at every  $t \in \mathbb{N}$ . Let  $\mathcal{G}_{\text{init}}^c$  denote the initial measurement graph that describes the neighbor relations used by the nodes to implement the algorithm, and let  $\mathcal{L}_c$  and  $\mathbf{b}_c$  be as defined in (3.12) and (3.16) for the pair  $(\mathcal{G}, \mathcal{G}_{\text{init}}^c)$ . Furthermore, assume that Diagonal Assumption 3.3.1 holds. Then, the Jacobi algorithm converges to the unique solution of  $\mathcal{L}_c \mathbf{x} = \mathbf{b}_c$  if and only if*

1. *for every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is a directed path in  $\mathcal{G}_{\text{init}}^c$  from at least one reference node to that node, and*

2. no communication edge in  $\mathcal{G}_{\text{init}}^c$  fails permanently, and no communication edge that is not in  $\mathcal{G}_{\text{init}}^c$  remains active infinitely often, i.e.,

$$\bigcap_{\ell=1}^{\infty} \bigcup_{t=\ell}^{\infty} \mathcal{G}^c(t) = \mathcal{G}_{\text{init}}^c. \quad (3.22)$$

When nodes and communication edges fail according to the i.i.d. failure model, if condition 1 above is satisfied, then the Jacobi algorithm converges to the unique solution of  $L_c \mathbf{x} = \mathbf{b}_c$  almost surely.  $\square$

*Cautionary remark:* All of the convergence results of the Jacobi algorithm in this dissertation, either with *asynchronous* iteration or with asymmetric communication, have been established for the special cases when Diagonal Assumption 3.3.1 is satisfied. A general convergence result is still an open problem. However, in the simulations described in Section 3.3.3, this assumption is violated but the algorithm is seen to converge. Such numerical evidence suggests that Diagonal Assumption 3.3.1 is perhaps required due to our proof technique, but can probably be relaxed.  $\square$

The following corollary about the correctness of the Jacobi algorithm follows immediately.

**Corollary 3.3.1.** *When Assumption 3.1.1 and the Diagonal Assumption 3.3.1 hold, the Jacobi algorithm implemented on a measurement graph  $\mathcal{G}$  and its associated time-varying communication graph  $\mathcal{G}^c(t)$  is correct if and only if the following conditions hold:*

1. The initial communication graph  $\mathcal{G}_{\text{init}}^c$  is symmetric.
2. No communication edge fails permanently and no communication edge that is not in  $\mathcal{G}_{\text{init}}^c$  remains active infinitely often.

*When the nodes and communication edges fail according to the i.i.d. failure model, and the first condition above is satisfied, then the Jacobi algorithm converges to the optimal estimates a.s.*  $\square$

Note that the condition of there being directed paths from reference nodes to the other nodes in Theorem 3.3.3 is automatically satisfied by  $\mathcal{G}_{\text{init}}^c$  being symmetric (see Assumption 3.1.1 to see why).

### 3.3.1.2 Convergence rate of the Jacobi algorithm

For establishing the convergence rate of Jacobi algorithm, we restrict our attention to the special case of symmetric communication without any communication faults, with the node variables being scalars (i.e.,  $k = 1$ ). As explained in Section 3.3.1.1, when Diagonal Assumption 3.3.1 is satisfied, the general case of vector-valued variables can be analyzed in terms of the scalar case. It follows from (3.18) that convergence rate of the Jacobi algorithm will depend on the spectrum of the Jacobi iteration matrix  $J$ .

We will need the following notation in presenting the convergence rate result. Let  $\lambda_{\min}(L)$  denote the minimum eigenvalue of the matrix  $L$ , and  $d_{\max}(P), d_{\min}(P)$  denote the maximum and minimum *weighted degrees* of the graph  $\mathcal{G}$ , i.e.,  $d_{\max}(P) := \max_j M_{j,j}$  and  $d_{\min}(P) := \min_j M_{j,j}$ . The dependence of  $d_{\max}, d_{\min}$  on  $P$  is used to emphasize that the edge weights are the inverse-variances which are specified by a function  $P : \mathcal{E} \rightarrow (0, \infty)$ . The proof is provided in Section 3.7.

**Theorem 3.3.4.** *Consider a pair of measurement and time-invariant communication graphs  $(\mathcal{G}, \mathcal{G}^c)$  that satisfies Assumption 3.1.1. Assume that  $k = 1$ ,  $\mathcal{G}^c$  is symmetric, and the Jacobi algorithm is implemented in a synchronous manner. For every  $0 < \epsilon < 1$ , the number of iterations  $n_{\text{iter}}(\epsilon)$  required so that*

$\varepsilon^{(i)} < \epsilon, \forall i > n_{\text{iter}}(\epsilon)$ , satisfies

$$\lfloor d_{\min}(P) \frac{|\log \epsilon|}{2\lambda_{\min}(L)} \rfloor \leq n_{\text{iter}}(\epsilon) \leq \lceil d_{\max}(P) \frac{|\log \epsilon|}{\lambda_{\min}(L)} \rceil. \quad \square$$

The advantage of the result above is that for a large class of graphs that are relevant to ad-hoc sensor networks, asymptotic bounds on  $\lambda_{\min}(L)$  can be obtained even without complete knowledge of the graph. We will obtain in Chapter 6 one such bound in terms of “effective resistances” in the graph. In addition, the ratios  $\lambda_{\min}(L)/d_{\max}$ ,  $\lambda_{\min}(L)/d_{\min}$  are closely related to the well-known algebraic connectivity of the unweighted graph  $\mathcal{G}$ , for which an extensive literature exists [81, 82].

### 3.3.2 Reducing error faster - flagged initialization

The preceding discussion shows that in measurement graphs with low algebraic connectivity, the Jacobi algorithm will take a large number of iterations before the error ratio  $\varepsilon^{(i)}$  becomes sufficiently small. Since large ad-hoc and sensor networks are expected to have low algebraic connectivity, one can expect that the number of iterations that the Jacobi algorithm takes before the error w.r.t. to the optimal estimate,  $\|\mathbf{e}^{*(i)}\|$ , is lower than a pre-specified value will be, in general, quite large.

There are two ways to reduce the error  $\|\mathbf{e}^{*(i)}\|$  : employ an algorithm with a faster convergence rate compared to Jacobi, and initialize the iterations with more accurate initial estimates. Devising a faster algorithm is postponed till Section 3.4. In this section we show how to reduce the error w.r.t. to the optimal estimate of the Jacobi algorithm by cleverly initializing the initial estimates. This scheme, called *flagged initialization*, does not require extra communication or expensive computation, and is also applicable to other algorithms. Indeed, the

flagged initialization will be used with the OSE algorithm to reduce its error.

After the deployment of the network, the reference nodes initialize their estimates to their known values, but all other nodes initialize their estimates to  $\infty$ , which serves as a flag to declare that these nodes do not have a good estimate of their variables. Subsequently, in its estimate updates, each node includes in its 1-hop subgraph only those nodes that have finite estimates. If none of their neighbors has a finite estimate, then the node keeps its estimate at  $\infty$ . In the beginning, only the references have a finite estimate. In the first iteration, the 1-hop neighbors of the references can compute finite estimates, whereas in the second iteration, the 2-hop neighbors of the references can also obtain finite estimates and so forth until all nodes have finite estimates. Flagged initialization affects only the initial stage of the algorithm, and thus does not affect its correctness and the rate at which the error ratio  $\varepsilon^{(i)}$  is reduced with iteration number  $i$ .

### 3.3.3 Simulations of the Jacobi algorithm

We present a few numerical simulations to study the behavior of the Jacobi algorithm. First we simulate the algorithm with symmetric communication and synchronous operation.

In these simulations the node variables represent the physical position of sensors in the plane. All simulations refer to a network with 200 nodes that are randomly placed in the unit square (see Figure 3.8). Node 1, placed at the origin, is chosen as the single reference node. Pairs of nodes separated by a distance smaller than  $r_{\max} := 0.11$  are allowed to have noisy measurements of each others' relative range and bearing (see Figure 2.1). The range measurements are corrupted with zero-mean additive Gaussian noise with standard deviation  $\sigma_r = 0.15 r_{\max}$ , and

the angle measurements are corrupted with zero-mean additive Gaussian noise with standard deviation  $\sigma_\theta = 10$  deg. Assuming that the range and bearing measurement errors are independent and have variances independent of distance, consider a noisy measurement  $(r, \theta)$  of true range and angle  $(r_o, \theta_o)$ . Then it can be shown that the covariance matrix of the measurement  $\zeta_{u,v} = [r \cos \theta, r \sin \theta]^T$  is given approximately by

$$P_{u,v} = \begin{bmatrix} y_o^2 \sigma_\theta^2 + \sigma_r^2 \cos^2 \theta_o & -x_o y_o \sigma_\theta^2 + \frac{\sigma_r^2}{2} \sin(2\theta_o) \\ -x_o y_o \sigma_\theta^2 + \frac{\sigma_r^2}{2} \sin(2\theta_o) & x_o^2 \sigma_\theta^2 + \sigma_r^2 \sin^2 \theta_o \end{bmatrix}, \quad (3.23)$$

where  $x_o = r_o \cos \theta_o$  and  $y_o = r_o \sin \theta_o$ . Assuming that the scalars  $\sigma_r, \sigma_\theta$  are provided a priori to the nodes, a node can estimate this covariance by using the measured  $r$  and  $\theta$  in place of their unknown true values. Since the covariances are not diagonal and since distinct measurements have distinct covariances, this example does not satisfy the assumptions for which the OSE algorithm is guaranteed to converge. The locations estimated by the centralized optimal estimator are shown in Figure 3.8, together with the true locations.

In reporting simulation results, we plot the normalized error vs. iteration number, where the normalized error is defined as

$$\frac{\|\hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*\|}{\|\hat{\mathbf{x}}^*\|}. \quad (3.24)$$

Recall that  $\hat{\mathbf{x}}^{(i)}$  is the vector of estimates at the  $i^{\text{th}}$  iteration and  $\hat{\mathbf{x}}^*$  is the optimal estimate.

Figure 3.9(a) compares the normalized error as a function of iteration number for the Jacobi algorithm, with and without flagged initialization. The straight lines in the log-scaled graph indicate the exponential convergence of the algorithm. The figure shows the dramatic improvement achieved with the flagged initialization scheme. With flagged initialization, the Jacobi algorithm can estimate the node

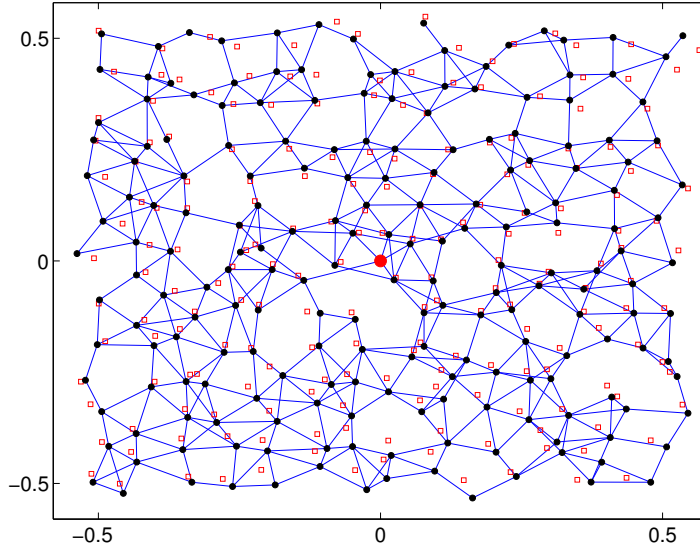
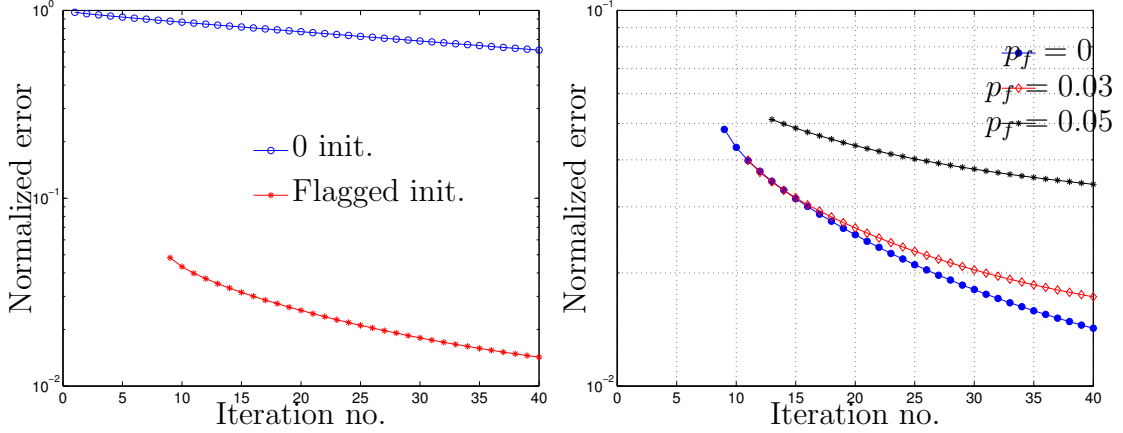


Figure 3.8. A measurement graph created by an ad-hoc sensor network with 200 nodes distributed randomly in a unit square area. The edges of the measurement graph are shown as line segments connecting the node positions, which are shown as black dots. Two nodes with an edge between them are provided with a measurement of their relative positions in the plane. The red squares are the positions estimated by the (centralized) optimal estimator. A single reference node is located at the origin.

positions within 5% of the optimal estimate after only 9 iterations.

Figure 3.9(b) shows the performance of the Jacobi algorithm with flagged initialization under i.i.d. communication link failure. Every symmetric communication link is allowed to fail (independently of all other links) with probability  $p_f$  at every iteration, and no node is allowed to fail. Therefore, in the terminology of Theorem 3.3.3,  $p = p_f$ ,  $q = 0$ . Note that although the communication graph  $\mathcal{G}_{\text{init}}$  is symmetric, the communication failures need not be. A communication edge  $(u, v)$  can fail independently of the communication edge  $(v, u)$ . Three values of the failure probabilities are tested. Not surprisingly, higher failure rates result in

slower convergence. Note that Theorem 3.3.3 guarantees the convergence of the error to 0 is guaranteed with link and node failure only when Diagonal Assumption 3.3.1 is satisfied. In these simulations the assumption was not satisfied (see the covariances in (3.23)), still the algorithm is seen to converge to the optimal estimates.



(a) Without and with flagged initialization. (b) Effect of communication link failure.

Figure 3.9. Simulation results (normalized error  $\frac{\|\hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*\|}{\|\hat{\mathbf{x}}^*\|}$  vs. iteration number) of Jacobi algorithm with symmetric communication. (a) With and without flagged initialization. “0 init.” means all the node estimates were initialized to 0. The communication graph is symmetric and time-invariant (no faults). (b) Effect of communication-link failures. All the simulations in case (b) are carried out with flagged initialization. Note that although the communication graph  $\mathcal{G}_{\text{init}}$  is symmetric, the communication failures need not be. A communication edge  $(u, v)$  can fail independently of the communication edge  $(v, u)$ .

Now we present numerical evidence that the Jacobi algorithm indeed converges to a non-optimal estimate when communication is asymmetric. The Jacobi algorithm is simulated for the measurement and communication graph pair shown in

Figure 3.1. Simulation results are shown in Figure 3.10. At every iteration of the simulation, every communication edge was allowed to fail with a probability of 0.2, independent of all other edges, i.e.,  $p = 0.2$  and  $q = 0$ . The Figure validates the predictions of Theorem 3.3.3 and Lemma 3.3.1: the estimate converges to the predicted value  $\hat{\mathbf{x}}^\infty$  but not to the optimal estimate  $\mathbf{x}^*$ .

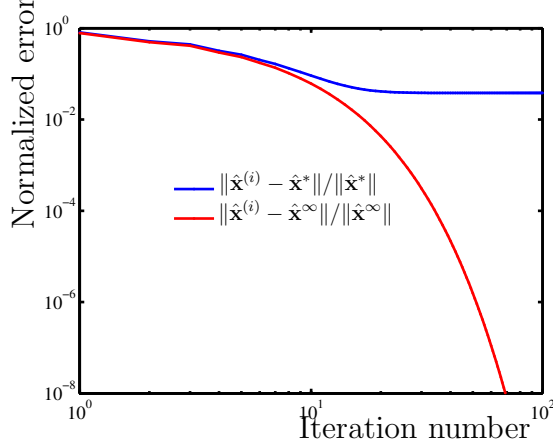


Figure 3.10. Simulation results on the convergence of the Jacobi algorithm with asymmetric communication. The simulation was conducted for the measurement graph  $\mathcal{G}_1$  and communication graph  $\mathcal{G}_1^c$  shown in Figure 3.1. Each communication edge was allowed fail at each iteration, with a probability of 0.2, independent of all other edges. The algorithm converges to an unbiased estimate  $\hat{\mathbf{x}}^\infty$  whose variance is larger than the that of the BLU estimate.

### 3.4 The overlapping subgraph estimator(OSE) algorithm

In this section we describe a distributed algorithm to compute the optimal estimates that has a faster convergence rate compared to the Jacobi algorithm.

In spite of the advantages of the Jacobi algorithm discussed above, such as scalability, convergence, correctness under mild assumptions, robustness to temporary failures, it has a significant weakness, namely, its slow convergence rate (see the discussion following Theorem 3.3.4).

It may be possible to improve the convergence rate by using other iterative techniques such as Gauss – Siedel, SOR or the conjugate gradient [75] methods, or even by preconditioning, but any such improvement will come at the cost of increased communication. In ad-hoc wireless networks, the primary source of energy consumption is communication [10], while much less energy is consumed for computation [11]. Therefore the challenge is to devise an algorithm that achieves faster convergence compared to the Jacobi algorithm with no, or minimal, increase in communication. The *overlapping subgraph estimator (OSE)* algorithm described in this section achieves these objectives. It also retains the scalability and robustness properties of the Jacobi algorithm.

*We will consider only the symmetric communication case in describing the OSE algorithm.* All the analysis and simulation of the OSE algorithm will be done under the assumption that inter-node communication is symmetric. A general analysis of the algorithm with asymmetric case is a subject of future research.

The OSE algorithm can be thought of as an extension of the Jacobi algorithm, in which individual nodes utilize larger subgraphs to improve their estimates. To understand how, suppose that each node broadcasts to its 1-hop neighbors not only its current estimate, but also all of the latest estimates that it received from its 1-hop neighbors. Note that since we have assumed symmetric communication between nodes, the 1-hop communication neighbors and 1-hop measurement neighbors are identical, which we refer to as the 1-hop neighbors. In the absence of drops, at the  $i$ th iteration step each node  $u$  has the estimates  $\hat{x}_v^{(i)}$  for its 1-hop

neighbors  $v \in \mathcal{N}_u(1)$  as well as the (older) estimates  $\hat{x}_v^{(i-1)}$  for its 2-hop neighbors  $v \in \mathcal{V}_u(2) \setminus \mathcal{N}_u(1)$ .

The reason that we don't attempt a complete analysis of the OSE algorithm in the presence of asymmetric communication, apart from the mathematical difficulty in carrying out such an analysis, is that – and it will be shown in Section 3.5 – it is impossible to ameliorate some of the detrimental effects of asymmetric communication by using the OSE algorithm in place of Jacobi.

Under the information exchange scheme described above, at the  $i^{\text{th}}$  iteration each node  $u$  has estimates of all of the node variables of the nodes in the set  $\mathcal{V}_u(2)$  consisting of all of its 1-hop and 2-hop measurement neighbors. In the OSE algorithm, each node updates its estimate using the *2-hop subgraph*  $\mathcal{G}_u(2) = (\mathcal{V}_u(2), \mathcal{E}_u(2))$  of  $\mathcal{G}$  centered at  $u$ , with edge set  $\mathcal{E}_u(2)$  consisting of all of the edges of the measurement graph  $\mathcal{G}$  that connect elements of  $\mathcal{V}_u(2)$ . For this estimation problem, node  $u$  takes as references the node variables of its 2-hop neighbors. The gain in convergence speed with respect to the Jacobi algorithm comes from the fact that the 2-hop subgraph  $\mathcal{G}_u(2)$  contains more edges than the 1-hop subgraph  $\mathcal{G}_u(1)$ . The *OSE algorithm* can be summarized as follows:

1. Each node  $u \in \mathcal{V}$  picks an arbitrary initial estimate  $\hat{x}_v^{(-1)}$  of the node variable  $x_v$  of each of its 2-hop neighbors  $v \in \mathcal{V}_u(2) \setminus \mathcal{V}_u(1)$ . These estimates need not be consistent across different nodes.
2. At the  $i$ th iteration, each node  $u \in \mathcal{V}$  assumes that the estimates  $\hat{x}_v^{(i-2)}$  of the node variables  $x_v$  of its 2-hop neighbors that it received through its 1-hop neighbors are correct and solves the corresponding optimal estimation problem associated with the 2-hop subgraph  $\mathcal{G}_u(2)$ . In particular, each node

$u$  solves the linear equations  $\mathcal{L}_{u,2}\mathbf{y}_u = \mathbf{b}_u$ , where  $\mathbf{y}_u$  is a vector of node variables that correspond to the nodes in its 1-hop subgraph  $\mathcal{G}_u(1)$ , and  $\mathcal{L}_{u,2}, \mathbf{b}_u$  are defined for the subgraph  $\mathcal{G}_u(2)$  as  $\mathcal{L}, \mathbf{b}$  are for  $\mathcal{G}$  in (2.6). After this computation, node  $u$  updates its estimate as  $\hat{x}_u^{(i+1)} \leftarrow \lambda y_u + (1 - \lambda)\hat{x}_u^{(i)}$ , where  $0 < \lambda \leq 1$  is a pre-specified design parameter and  $y_u$  is the variable in  $\mathbf{y}_u$  that corresponds to  $x_u$ . The new estimate  $\hat{x}_u^{(i+1)}$  as well as the estimates  $\hat{x}_v^{(i)}$  previously received from its 1-hop neighbors  $v \in \mathcal{V}_u(1)$  are then broadcast to all of its 1-hop neighbors.

3. At the end of the  $i$ th iteration, each node  $u$  then listens for the broadcasts from its 1-hop neighbors and uses them to update its estimates for the node variables of all of its 2-hop neighbors. Once all updates are received a new iteration can start.

As in the case of the Jacobi algorithm, the termination criteria vary depending on the application, and nodes use measurements and covariances obtained initially for all future time. Figure 3.11 shows a 2-hop subgraph used by the OSE algorithm.

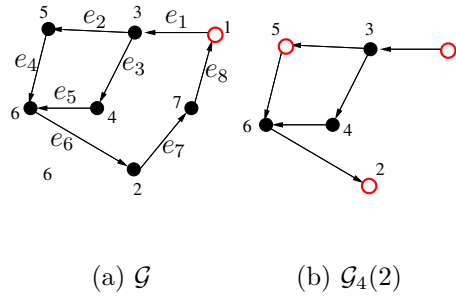


Figure 3.11. (a) A measurement graph  $\mathcal{G}$  with node 1 as reference, and (b) a 2-hop subgraph  $\mathcal{G}_4(2)$  centered at node 4. While running the OSE algorithm, node 4 treats nodes 1, 5, and 2 as reference nodes in the subgraph  $\mathcal{G}_4(2)$  and solves for the unknowns  $x_3, x_4$ , and  $x_6$ .

The previous description assumes that communication is synchronous and that each node receives broadcasts from all of its neighbors. However, as in the Jacobi algorithm, the OSE algorithm can be implemented in an asynchronous manner to make the algorithm robust to imperfect synchronization and link failures. A timeout mechanism can be used for this purpose, in which each node resets a timer as it broadcasts its most recent estimates. When this timer reaches a pre-specified timeout value, the node initiates a new iteration, regardless of whether or not it received messages from all of its 1-hop neighbors. If a message is not received from one of its neighbors, the node uses the data most recently received from that neighbor for the next iteration.

*Remark 3.4.1.  $h$ -hop OSE* One can also design a  $h$ -hop OSE algorithm by letting every node utilize a  $h$ -hop subgraph centered at itself, where  $h$  is an (small) integer. The resulting algorithm is a straightforward extension of the 2-hop OSE just described, except that at every iteration, individual nodes have to transmit to their neighbors larger amounts of data than in 2-hop OSE, potentially requiring multiple packet transmissions at each iteration. In practice, this added communication cost limits the allowable value of  $h$ .  $\square$

The next result establishes the correctness of the OSE algorithm.

**Theorem 3.4.1.** *Imagine a pair of measurement and initial communication graphs  $(\mathcal{G}, \mathcal{G}_{\text{init}}^c)$  satisfying assumption 3.1.1. When the communication graph  $\mathcal{G}^c(t)$  is symmetric at every  $t \in \mathbb{N}$ , and Diagonal Assumption 3.3.1 is satisfied, and no node or communication edge fails permanently, i.e.,  $\bigcap_{\ell=1}^{\infty} \bigcup_{t=\ell}^{\infty} \mathcal{G}^c(t) = \mathcal{G}_{\text{init}}^c$ , then the OSE algorithm converges to the optimal estimate.*  $\square$

*Proof of Theorem 3.4.1.* In the special case  $\lambda = 1$ , the OSE algorithm becomes the same as the Asynchronous Weighted Additive Schwarz (AWAS) method [83].

In that case, Theorem 3.1 in [83] states that if  $L^{-1} \succeq 0$  and some weak regularity condition holds, then the AWAS method converges for every initial condition. Under the assumptions, it follows from Lemma 3.3.2 that  $L$  is a non-singular M-matrix, and therefore  $L^{-1} \succ 0$  [78]. The splitting  $L = M - N$  is called weak regular if  $M^{-1} \succeq 0$  and  $M^{-1}N \succeq 0$ , which is satisfied in our case. The regularity condition required in [83] is actually not on the splitting  $M - N$  but on a number of splittings that every node can be thought of as applying in its local processor. We refrain from repeating the tedious details, but it is straightforward to check that the OSE algorithm satisfies the weak regularity conditions in Theorem 3.1 of [83]. This guarantees convergence of the AWAS method, and therefore of the OSE algorithm. When  $\lambda < 1$ , the proof technique of Theorem 3.1 in [83] can be adapted to prove again that the OSE algorithm converges. Since the proof is extremely long and tedious, yet only a minor generalization of the results in [83], it is not provided here. The complete proof is, however, available in [66].

Note that the inverse-positivity of M-matrices, and specifically of  $L$  was used above, but not in proving convergence of the Jacobi algorithm. As in case of the Jacobi algorithm, the Assumption 3.3.1 can probably be relaxed for the OSE algorithm's correctness. In the simulations described below, this assumption is violated but the algorithm is seen to converge.

#### 3.4.0.1 Modified EPA

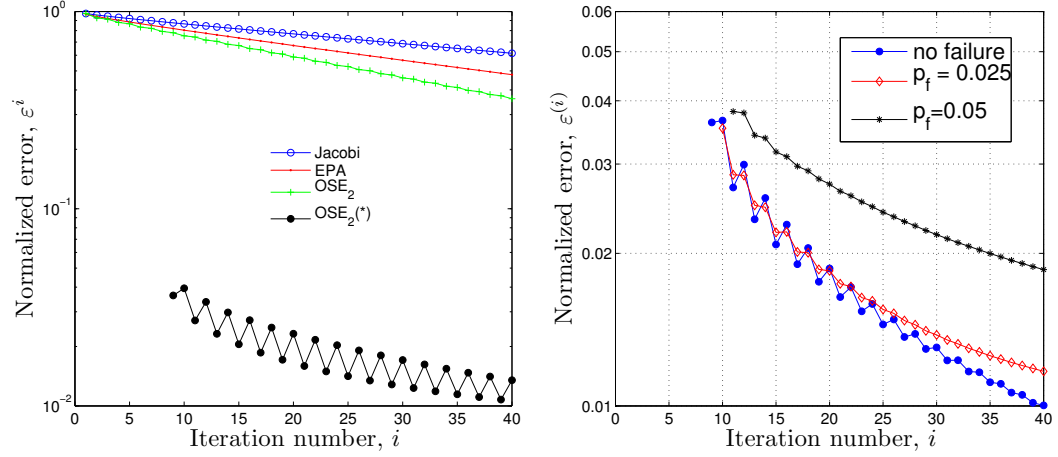
The Embedded Polygon Algorithm (EPA) proposed in [69] can be used for iteratively solving (2.6); since it is essentially a block – Jacobi method of solving a system of linear equations, where the blocks correspond to non-overlapping polygons. The special case when the polygons are triangles has been extensively

studied in [69]. We will not include here the details of the algorithm, including triangle formation in the initial phase, the intermediate computation, communication and update. The interested reader is referred to [69]. It is not difficult to adapt the algorithm in [69] to the problem considered here. We have implemented the modified EPA algorithm (with triangles as the embedded polygons) and compared it with both Jacobi and OSE. Results are presented in section 3.4.2.

### 3.4.1 Simulations of the OSE algorithm

In this section, we present numerical simulations to illustrate the performance of the OSE algorithm, and compare its convergence rate with that of Jacobi numerically. All the simulations with the OSE algorithm are done for the measurement graph shown in Figure 3.8. The construction of the measurement graph along with the measurements and their associated error covariances are described in Section 3.3.3.

Figure 3.12(a) compares the normalized error as a function of iteration number for the three algorithms discussed in this paper - Jacobi, EPA and the OSE. Two versions of OSE were tested, 2-hop and 3-hop. It is clear from this figure that the OSE outperforms both Jacobi and modified EPA. As the figure shows, drastic improvement was achieved with the flagged initialization scheme. With flagged initialization, the 2-hop OSE algorithm can estimate the node positions within 3% of the optimal estimate after only 9 iterations. For the flagged OSE, the normalized error is not defined till iteration number 8, since some nodes had no estimate of their positions till that time. Figure 3.12(b) shows the performance of the 2-hop OSE algorithm with flagged initialization under two different link-failure probabilities. Not surprisingly, higher failure rates result in slower convergence.



(a) Jacobi and OSE.

(b) 2-hop OSE with link failures.

Figure 3.12. (a) Performance comparison between the Jacobi algorithm and the overlapping subgraph estimator (OSE) algorithm without link failures. The normalized error is defined as  $\frac{\|\hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*\|}{\|\hat{\mathbf{x}}^*\|}$ , where  $\hat{\mathbf{x}}^{(i)}$  is the vector of estimates at the  $i$ -th iteration and  $\hat{\mathbf{x}}^*$  is the optimal estimate. Except for the case with flagged initialization, all of the simulations are run with all initial estimates of node variables set to 0. For the flagged OSE, the normalized error can be defined only after iteration number 8 because until then not all nodes have valid (finite) estimates. (b) Performance of 2-hop OSE with link failures. All simulations are run with flagged initialization. Two different failure probabilities are compared with the case of no failure. With higher probability of failure, performance degrades but the error is seen to decrease with iteration count even with large failure probabilities.

### 3.4.2 Energy cost comparison

For ad-hoc wireless network applications, the primary metric for comparison between the algorithms described above is not the number of iterations required to drive the error below a certain value, but the average energy consumed in order to do so. The reason is that in ad-hoc wireless sensor networks, one of the main challenges is to keep the network functional for an extended period of time in spite of the small battery life of the sensors [84]. Reducing energy consumption is therefore critical.

The OSE algorithm converges faster than both Jacobi and EPA. However, faster convergence is achieved at the expense of each node sending and processing more data. One may then ask whether there is a significant advantage to using the OSE algorithm. However, the energy cost of sending additional data can be negligible due to the complex dependence of energy consumption in wireless communication on radio hardware, underlying PHY and MAC layer protocols, network topology and a host of other factors.

Investigation into energy consumption of wireless sensor nodes has been rather limited. Still, we can get an idea of which parameters are important for energy consumption from the studies reported in [85–87]. It is reported in [87] that for very short packets (in the order of 100 bits), transceiver startup dominates the power consumption; so sending a very short message offers no advantage in terms of energy consumption over sending a somewhat longer message. In fact, in a recent study of dense network of IEEE 802.15.4 wireless sensor nodes, it is reported in transmitted energy per bit in a packet *decreases monotonically* upto the maximum payload [85]. One of the main findings in [86] was that in highly contentious networks, “transmitting large payloads is more energy efficient”. On

the other hand, receive and idle mode operation of the radio is seen to consume as much energy as the transmit mode, if not more [88]. Thus, the number of packets sent and received appear to be a better measure to predict energy consumption than the number of bits.

Due to the reasons outlined above, we take the number of packets transmitted and received by a node as a measure of its energy consumption. Let  $N_{tx}^{(i)}(u)$  be the number of packets a node  $u$  transmits to its neighbors during the  $i$ th iteration. The energy  $E^{(i)}(u)$  expended by  $u$  in sending and receiving data during the  $i$ th iteration is computed by the following formula:

$$E^{(i)}(u) = N_{tx}^{(i)}(u) + \frac{3}{4} \sum_{v \in \mathcal{N}_u} N_{tx}^{(i)}(v), \quad (3.25)$$

where  $\mathcal{N}_u$  is the set of neighbors of  $u$ . The factor  $3/4$  is chosen to account for the ratio between the power consumptions in the receive mode and the transmit mode. Our choice is based on values reported in [85] and [89]. The average energy consumption  $\bar{E}(\epsilon)$  is the average (over nodes) of the total of energy consumed among all the nodes till the normalized error reduces to  $\epsilon$ . For simplicity, eq. (3.25) assumes synchronous updates and perfect communication (no retransmissions). When packet transmission is unsuccessful, multiple retransmissions maybe result, making the resulting energy consumption a complex function of the parameters involved [85, 86].

In one iteration of the Jacobi algorithm, a node needs to broadcast its own estimate, which consists of  $k$  real numbers. Recall that  $k$  is the dimension of the node variables. Assuming a 32 bit encoding, that amounts to  $4k$  bytes of data. In the OSE algorithm, a node with  $d$  neighbors has to broadcast data consisting of  $4d$  bytes for its neighbors' IP addresses,  $4k(d+1)$  bytes for the previous estimates of itself and its neighbors, and  $3d$  bytes for time stamps of those estimates. This

leads to a total of  $(7 + 4k)d + 4k$  bytes of data, and consequently the number of packets in a message becomes

$$N_{tx}(u) = \lceil \frac{(7 + 4k)d + 4k}{max\_databytes\_pkt} \rceil, \quad (3.26)$$

where *max\_databytes\_pkt* is the maximum number of bytes of data allowed in the payload per packet. We assume that the maximum data per packet is 118 bytes, as per IEEE 802.15.4 specifications [90]. For comparison, we note that the number of bytes in a packet transmitted by MICA motes can vary from 29 bytes to 250 bytes depending on whether B-MAC or S-MAC is used [91]. If the number of data bytes allowed is quite small, OSE may require multiple packet transmission in every iterations, making it more expensive.

The average energy consumption  $\bar{E}$  of the three algorithms – Jacobi, modified EPA and 2-hop OSE – are compared in Figure 3.13. Flagged initialization was used in all three algorithms. To compute the energy consumption for the 2-hop OSE, we apply (3.26) with  $k = 2$  and *max\_databytes\_pkt* = 118 to get  $N_{tx}(u) = \lceil (15d_u + 8)/118 \rceil$ . The average node degree being 5, the number of packets broadcasted per iteration in case of the OSE algorithm was 1 for almost all the nodes. For Jacobi, the number of packets broadcasted at every iteration was 1 for every node. For the modified EPA algorithm, the number of packets in every transmission was 1 but the total number of transmissions in every iteration were larger (than Jacobi and OSE) due to the data exchange required in both the EPA update and EPA solve steps (see [69] for details). The normalized error against the average (among all the nodes) total energy consumed  $\bar{E}$  is computed and plotted in Figure 3.13. Comparing the plots one sees that for a normalized error of 1%, the OSE consumes about 70% of the energy consumed by modified EPA and 60% of that by Jacobi. As lower errors are demanded, the difference becomes more drastic: to achieve a normalized error of 0.8%, OSE needs only 60%

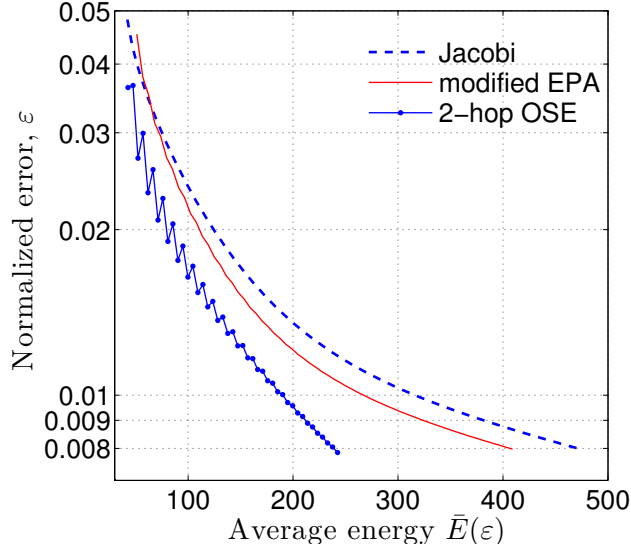


Figure 3.13. The normalized error  $\frac{\|\hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*\|}{\|\hat{\mathbf{x}}^*\|}$  vs. average energy consumption of 2-hop OSE, modified EPA and Jacobi with broadcast communication. Flagged initialization was used in all the three algorithms.

of the energy consumed by EPA and about half of that by Jacobi.

Note that the energy consumption benefits of OSE become more pronounced as higher accuracy is demanded, but less so for low accuracy. This feature is due to the flagged initialization, which accounts for almost all the error reduction in the first few iterations. Note that the energy savings in OSE will occur only if the extra data can be packed into a small number of packets. In such cases, the OSE algorithm is advantageous compared to the Jacobi algorithm because OSE requires a smaller number of iterations – and therefore a smaller number of messages – compared to Jacobi to achieve a desired error tolerance, resulting in lower energy consumption and increased network life.

## 3.5 Effect of asymmetric communication

In Section 3.3 we saw that if the communication graph is asymmetric, the Jacobi algorithm does not converge to the optimal estimate. This raises two questions:

1. Is it possible to design a distributed algorithm that converge to the optimal estimate even in the presence of asymmetric communication?
2. How sensitive is the Jacobi algorithm (or some other distributed algorithm) to the level of asymmetry? In other words, does increasing the level of asymmetry make the variance of the estimates (that the algorithm converges to) larger?

### 3.5.1 An impossibility result

The answer to the first question is, no. This can be seen by the example shown in Figure 3.14, where the reference variable is  $x_1 = 0$  and the measurement error variances are equal. Consider the case when the communication graph (shown in Figure 3.14) is time-invariant. It satisfies the conditions of Theorem 3.3.2, and therefore the Jacobi algorithm converges. However, due to the asymmetry in the communication graph, the limiting estimate will be different from the optimal estimate. It is clear from the figure that due to the information flow structure imposed by the communication graph, node 2 will only have information of the reference variable, which is 0, and the measurement  $\zeta_{12}$ . The optimal estimate of  $x_2$  is, however, a combination of all three measurements:  $\hat{x}_2^* = -\frac{2}{3}\zeta_{12} - \frac{1}{3}(\zeta_{13} - \zeta_{32})$ . Clearly no distributed algorithm can converge to the optimal estimate, since information on  $\zeta_{13}$  will never reach node 2. Even if nodes are allowed to transmit

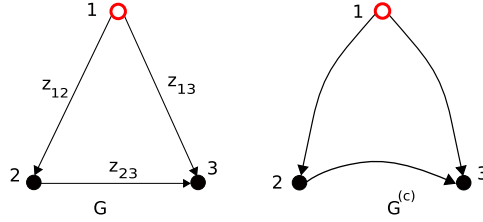


Figure 3.14. A measurement and communication graph pair in which it is impossible for any distributed algorithm to converge to the optimal estimate. The difficulty arises from the asymmetry in the communication graph that prevents information of the relative measurements on certain edges from reaching certain nodes.

their neighbors' information in addition to their own, as done in the OSE algorithm, similar examples can be constructed that shows the impossibility of optimal estimation in the presence of communication asymmetry.

It is important to keep in mind that when we say no distributed algorithm can converge to the optimal estimate when communication asymmetric, we are talking about algorithms that satisfy the Constraint 3.1.

### 3.5.2 More measurements need not reduce error

Another important effect of asymmetric communication is that using more measurements need not lead to more accurate estimates of all node variables - the variance of some of the node variables' estimation error can in fact increase. When communication is symmetric, the Jacobi algorithm converges to the optimal estimate. The optimal estimate has the property that its estimation error variance can only decrease upon using more measurements. This follows from the so-called Rayleigh's monotonicity law of effective resistances; see Theorem 4.6.1 in

Chapter 4 for details. We conclude that with symmetric communication, having more measurement edges, regardless of the associated error, produces more accurate (less variance) estimates when either the Jacobi or the OSE algorithm is used. However, the presence of asymmetry in the communication graph destroys this monotonicity. We illustrate this effect of asymmetry with a particularly troubling example, where the addition of a measurement edge causes the error variances of *all* the node estimates to increase.

Figure 3.15 shows two measurements graphs  $\mathcal{G}_1$  and  $\mathcal{G}_2$  and their associated communication graphs  $\mathcal{G}_1^c$  and  $\mathcal{G}_2^c$ . The measurement graph  $\mathcal{G}_1$  contains all the nodes and edges of the measurement graph  $\mathcal{G}_2$ . Similarly,  $\mathcal{G}_1^c$  contains all the nodes and edges in  $\mathcal{G}_2^c$ . Every measurement error variance in both the measurements graphs is unity. The estimation error variances of the limiting estimates  $\hat{\mathbf{x}}^\infty$  computed from Lemma 3.3.1 are shown alongside the graphs. It is clear from the variances that the estimates in  $\mathcal{G}_1$  are poorer than those in  $\mathcal{G}_2$ , even though  $\mathcal{G}_1$  contains more measurements than  $\mathcal{G}_2$ .

### 3.6 Comments and open problems

Among the questions left unanswered in this Chapter, perhaps the most important ones are on distributed algorithms for estimating time-varying node variables and establishing asynchronous convergence results for the Jacobi and OSE algorithms in the general case of arbitrary positive definite edge covariances (i.e., without Diagonal Assumption 3.3.1).

Distributed estimation of node variables that are changing with time, i.e., that have dynamics, is a challenging problem. It is equivalent to distributed Kalman filtering, which is an open research problem.

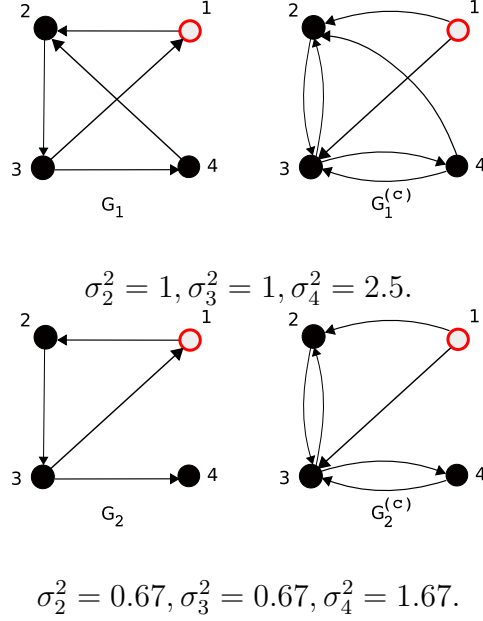


Figure 3.15. An example of more measurements reducing the estimation accuracy of the Jacobi algorithm when communication is asymmetric. The figure shows two measurements graphs  $\mathcal{G}_1$  and  $\mathcal{G}_2$ , and their associated communication graphs  $\mathcal{G}_1^c$  and  $\mathcal{G}_2^c$ . The variance of every measurement error is 1 for both the measurement graphs. The estimation error variances of the limiting estimate (that the Jacobi algorithm converges to) computed from Lemma 3.3.1 are shown alongside the graphs. Even though  $\mathcal{G}_2 \subset \mathcal{G}_1$  and  $\mathcal{G}_2^c \subset \mathcal{G}_1^c$ , the resulting estimation error variances are still higher in  $(\mathcal{G}_1, \mathcal{G}_1^c)$  than in  $(\mathcal{G}_2, \mathcal{G}_2^c)$ .

To prove convergence of the asynchronous version of the algorithms (both Jacobi and OSE), we had to assume a special structure of the measurement error covariance matrices (see Diagonal Assumption 3.3.1). The reason is the following. Consider a linear system of the form  $Lx = b \in \mathbb{R}^n$  with  $L \in \mathbb{R}^{n \times n}$  non-singular, and let  $L = M - N$  be a splitting of  $L$ , i.e.,  $M$  is non-singular. Define the iteration operator

$$J : \mathbb{R}^n \rightarrow \mathbb{R}^n, x \rightarrow M^{-1}(Nx + b).$$

For the synchronous iteration to converge to the solution  $L^{-1}b$ , we need  $\rho(J) < 1$ . The *asynchronous* iteration (corresponding to the synchronous one above) will also converge to the solution for every initial condition, if  $\rho(|J|) < 1$ , where  $|J|$  represents the matrix obtained by replacing all the entries of  $J$  with their absolute values [92]. This condition is also necessary, i.e., if  $\rho(|J|) \geq 1$ , then there exists an initial condition and sequence of communication faults for which the asynchronous iteration will not converge to the solution of  $Ax = b$  [92]. The reader is advised to see [92] and references therein for a review and historical perspective on the subject of asynchronous parallel iterations. The Diagonal Assumption 3.3.1 allowed us to reduce the problem to the special case of scalar valued node variables, i.e.,  $k = 1$ . The case  $k = 1$  offered us two distinct advantages, one, the iteration matrix  $J = M^{-1}N$  turned out to be non-negative, so we only had to prove  $\rho(J) < 1$ . The second advantage is that when  $k = 1$ ,  $M - N$  turned out to be an M-matrix, which allowed us to exploit the available results in the extensive literature on M-matrices and convergence of parallel iterative methods to show that the OSE algorithm converges.

In the general case the edge weights are positive-definite matrices and not positive scalars,  $J = \mathcal{M}^{-1}\mathcal{N}$  is not a non-negative matrix and  $\mathcal{L} = \mathcal{M} - \mathcal{N}$  is not a M-matrix. In fact, for the measurement network described in Section 3.3.3 for which numerical simulations were conducted,  $\rho(|J|) > 1$ . Therefore an asynchronous scheme will not converge in general. However, in both the algorithms the components of a node variable are always transmitted together, so the asynchronous iterations that are of interest to us have a special structure. Moreover, the special structure of  $\mathcal{L}$  indicates that thinking of it as a “block” M matrix, the Jacobi algorithm still might lead to a provably convergent asynchronous iteration. Such generalization of M-matrices have in fact been attempted, though

in a quite restrictive sense in most cases [93–96]. We believe there is hope for proving asynchronous iteration convergence with block M-matrices, but in order to do that a research program in generalizing M-matrices to block M-matrices has to be undertaken first. So significant technical hurdles remain.

Another useful research direction is the investigation of convergence rates with random communication faults. For this, second moment convergence has to be established first. Convergence rate on the second moment of the error with random link failures, even for simple failure distributions will be quite useful to the practitioner. Since the motivation for parallel iterative methods has traditionally been solution of large problems in clusters of powerful machines, which are connected over a wired network, asynchrony usually comes from delays in the wired network and varying processor speeds. As a result, in the vast literature on parallel iterative methods, analysis of convergence with random faults is rare (one notable exception being the work of Strikwerda [97]). With the recent interest in distributed computation in wireless networks, analysis of parallel iterative methods with random communication faults may be quite useful.

No analytical results on the convergence rate of the OSE algorithm were obtained here. It was shown through simulations to converge faster. Establishing convergence rate of the Weighted Additive Schwarz method, to which OSE is closely related, is recognized to be quite challenging [83]. Obtaining convergence rate of the OSE algorithm is therefore a challenging open problem, but one that is also of interest to a wider community.

## 3.7 Proofs

In the proofs, we will use properties of non-negative matrices and M - matrices. First we show the following:

**Proposition 3.7.1.** *The matrix  $L_c$  defined in (3.12) for the graph pair  $(\mathcal{G}, \mathcal{G}^c)$ , for the special case of  $k = 1$ , is an M-matrix as long as Assumption 3.1.1 is satisfied.  $\square$*

*Proof of Proposition 3.7.1.* In the case  $k = 1$ , the measurement error covariance  $P_e$  on edge  $e$  is simply a variance  $\sigma_e^2$ . Setting

$$s := \max_{u \in \mathcal{V}} [D]_{u,u},$$

$$B := sI - L_c,$$

and applying the Gerschgorin circle theorem [75], we conclude that  $\rho(B) \leq s$ . This proves that  $L_c$  is an M-matrix.  $\blacksquare$

**Proposition 3.7.2 (Theorem 2.3( $\mathbb{N}_{45}$ ) of [78]).** *Let  $X$  be a matrix of class  $Z$ , that is, it is a real square matrix whose off-diagonal terms are non-positive. Then the following statements are equivalent:*

1. *there exist a representation  $X = K - Q$  with  $K^{-1} \succeq 0$ ,  $Q \succeq 0$  such that  $\rho(M^{-1}N) < 1$ .*
2.  *$X$  is a non-singular M-matrix.*  $\square$

*Proof of Lemma 3.3.2.* Consider the following discrete-time dynamical system:

$$\mathbf{e}^{(i+1)} = M^{-1}N\mathbf{e}^{(i)}, \tag{3.27}$$

where  $\mathbf{e}^{(i)} \in \mathbb{R}^{n_b}$ . Note that  $M$  is invertible by construction since it is a diagonal matrix and every diagonal element is positive, which is guaranteed by Assumption 3.1.1. Due to the structure of the matrices  $M^{-1}$  and  $N$ , (3.27) implies that in every iteration  $i$ , each node  $u \in \mathcal{V}$  computes its new state  $e_u^i$  as the weighted average of the states of those nodes  $v$  that have an edge  $(v, u)$  directed from  $v$  toward  $u$  in the graph  $\mathcal{G}^c$ . In other words, it is a distributed average-consensus algorithm where the reference nodes keep their values at 0, and the remaining nodes try to reach consensus by averaging with their neighbors. The system (3.27) satisfies the strict convexity assumption of [98]. Thus, from Theorem 2 of [98] we know that the system (3.27) is uniformly globally attractive with respect to the collection of equilibrium points (which in this case is  $\{0\}$ ) if and only if for every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is at least one reference node such that there is a directed path in  $\mathcal{G}^c$  from the reference node to  $u$ . Note that here we have used a slight specialization of the results in [98] to the case when one or more agents do not participate in the consensus algorithm but keep their values fixed. On the other hand, it follows from (3.27) that  $\mathbf{e}^{(i)} \rightarrow 0$  as  $i \rightarrow \infty$  if and only if  $\rho(M^{-1}N) < 1$ . Since  $L_c$  is an M-matrix, it follows from Proposition 3.7.2  $\mathcal{L}_c = M - N$  is non-singular if and only if for every node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ , there is at least one reference node such that there is a directed path in  $\mathcal{G}^c$  from the reference node to  $u$ . This proves the first statement of the theorem.  $\blacksquare$

*Proof of Theorem 3.3.3.* We only consider the case  $k = 1$  in the proof. The case for  $k > 1$  follows from Assumption 3.3.1, as explained earlier.

When no edge or node fails permanently, the Jacobi algorithm with time-varying communication qualifies as an *asynchronous iteration* as defined in [92, Defn. 2.2]. Now we use Theorem 4.1 from [92] which states that the *asynchronous iteration*

corresponding to the synchronous iteration

$$x \leftarrow M^{-1}(Nx + b)$$

converges to the solution of  $(M - N)x = b$ , if the following conditions are satisfied:

1.  $M$  is non-singular,
2.  $\rho(|M^{-1}N|) < 1$ , where, for a matrix  $X$ ,  $|X|$  denotes the matrix obtained by replacing every entry of  $X$  with its absolute value.

The first condition is satisfied by Assumption 3.1.1 as proved in Proposition 3.3.1. From the condition on existence on paths, it follows that  $\rho(M^{-1}N) < 1$  (see Lemma 3.3.2). Because of the non-negativity of  $M$  and  $N$ , which follows from Diagonal Assumption 3.3.1, we get  $\rho(|M^{-1}N|) < 1$ . This proves the first part of the theorem.

If the existence of path condition is violated, it follows from Lemma 3.3.2 that  $\rho(M^{-1}N) \geq 1$ , which means the algorithm will not converge. If a communication edge or node fails permanently, or a communication edge that is not in  $\mathcal{G}_{\text{init}}^c$  becomes active at a later time and is active infinitely often, then we can construct a new “initial” communication graph  $\mathcal{G}_2^c$  that includes that communication edge, and apply the arguments above to conclude that the Jacobi algorithm converges, but to the solution of  $L_{c2}\mathbf{x} = \mathbf{b}_{c2}$  that is defined for  $(\mathcal{G}, \mathcal{G}_2')$ . Assumption 3.1.1 ensures that  $\mathcal{L}_c$  changes if the communication graph  $\mathcal{G}_{\text{init}}^c$  is changed. Therefore, the Jacobi algorithm cannot converge to the solution of  $\mathcal{L}_c\mathbf{x} = \mathbf{b}_b$  if a communication edge that is not in  $\mathcal{G}_{\text{init}}^c$  becomes active at a later time and is active infinitely often.

To prove almost sure convergence in the presence of random failures, define the

events:

$$\mathcal{C} = \{ \text{The Jacobi algorithm converges} \}$$

$$\mathcal{E}_j^e = \{ \text{The communication edge } e \text{ is active in time } j \}$$

$$\mathcal{A}_j = \cap_{e \in \mathcal{E}^c} \mathcal{E}_j^e$$

$$\mathcal{A}_j \text{ i.o.} = \cap_{j=1}^{\infty} \cup_{\ell \geq j} \mathcal{A}_\ell$$

Occurrence of the event  $\mathcal{A}_j$  means all the communication edges were active in time  $j$ , i.e., no edge or node failed in that time. If  $\mathcal{A}_j$  occurs for infinitely many  $j$ 's, i.e., if  $\mathcal{A}_j$  i.o. occurs, then the algorithm converges, since two sufficient conditions mentioned above for the convergence of the asynchronous Jacobi algorithm are satisfied. Therefore,

$$P(\mathcal{C}) \geq P(\mathcal{A}_j \text{ i.o.}), \quad (3.28)$$

where  $P(\cdot)$  denotes probability. We will now show that the right hand side above is 1. Since  $\mathcal{A}_j$  occurs if and only if every communication edge and every node is active in time  $j$ , we have

$$P(\mathcal{A}_j) = (1-p)^m(1-q)^n, \quad \forall j.$$

which is a positive number (since  $p, q < 1$ ) that does not depend on  $j$ . It follows that

$$\sum_{j=1}^{\infty} P(\mathcal{A}_j) = \infty,$$

and moreover, the events  $\{\mathcal{A}_j\}$  are independent since the node or edge failures at a time step are independent of failures in the past and the future. Therefore, by the Borel-Cantelli lemma [99], we get

$$P(\mathcal{A}_j \text{ i.o.}) = 1.$$

It follows from (3.28) that

$$P(\mathcal{C}) = 1,$$

which shows that the Jacobi algorithm converges almost surely.  $\blacksquare$

*Proof of Theorem 3.3.4.* Recall the decomposition  $L_c = M - N$  with  $M$  diagonal. Due to the assumption of symmetric communication,  $L = L_c$  and the matrices  $L, M, N$  are symmetric. The iteration counter is updated simultaneously by all the nodes. The error in the  $i^{\text{th}}$  iteration,  $\mathbf{e}^{(i)} = \hat{\mathbf{x}}^{(i)} - \hat{\mathbf{x}}^*$  propagates as

$$\mathbf{e}^{(i)} = J^i \mathbf{e}^{(0)}, \text{ where } J := M^{-1}N.$$

Since the communication graph is symmetric, it follows from Theorem 3.3.2 that the Jacobi algorithm converges to the optimal estimates, and  $\rho(J) < 1$ . It follows from the above that

$$\frac{\|\mathbf{e}^{(i)}\|}{\|\mathbf{e}^{(0)}\|} \leq \rho(J)^i \Rightarrow \varepsilon^{(i)} \leq \rho(J)^i,$$

where the second inequality follows from the definition of the normalized error in (3.3). It is easy to see from the above that

$$n_{\text{iter}}(\epsilon) = \lceil \frac{|\log \epsilon|}{|\log \rho(J)|} \rceil. \quad (3.29)$$

From Theorem 5.6 of [78], we have

$$\rho(J) = \frac{\rho(L^{-1}N)}{1 + \rho(L^{-1}N)} = 1 - \frac{1}{1 + \rho(L^{-1}N)}. \quad (3.30)$$

Now,  $L^{-1}N = L^{-1}M - I$ , and the eigenvalues of  $L^{-1}M$  are the same as those of  $M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}$ , where  $M^{\frac{1}{2}}$  is the unique positive definite square root of the positive definite diagonal matrix  $M$ , we get

$$\begin{aligned} \rho(L^{-1}N) &= \rho(M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}) - 1 \\ &= \|M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}\| - 1, \end{aligned} \quad (3.31)$$

where  $\|\cdot\|$  represents the matrix 2-norm and the second equality follows from the matrix  $M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}$  being symmetric positive definite. From the definition of the matrix 2-norm, we have

$$\begin{aligned}\|M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}\| &= \max_{y \neq 0} \frac{y^T M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}y}{y^T y} \\ &= \max_{z \neq 0} \frac{z^T L^{-1}z}{z^T M^{-1}z},\end{aligned}$$

where  $z := M^{\frac{1}{2}}y$ . It is straightforward to see that for every vector  $z$ ,  $z^T M^{-1}z \geq \frac{1}{d_{\max}(P)}z^T z$ . Therefore,

$$\|M^{\frac{1}{2}}L^{-1}M^{\frac{1}{2}}\| \leq \max_{z \neq 0} \frac{z^T L^{-1}z}{z^T z} d_{\max}(P) = \frac{d_{\max}(P)}{\lambda_{\min}(L)},$$

where  $\lambda_{\min}(L)$  denotes the smallest eigenvalue of  $L$ . It follows from (3.31) that

$$\begin{aligned}\rho(L^{-1}N) &\leq \frac{d_{\max}(P)}{\lambda_{\min}(L)} - 1 \\ \Rightarrow \rho(J) &\leq 1 - \frac{\lambda_{\min}(L)}{d_{\max}(P)}. \quad (\text{from (3.30)})\end{aligned}$$

It follows from the Gerschgorin circle theorem [100, pg. 498] that all eigenvalues of  $L$  are less than  $d_{\max}$ , so  $\frac{\lambda_{\min}(L)}{d_{\max}(P)} < 1$ . Taking logarithm of both sides of the inequality above, and using  $\log(1-x) > -x$  for  $0 < x < 1$ , we get

$$|\log \rho(\mathcal{G})| \geq \frac{\lambda_{\min}(L)}{d_{\max}}.$$

Plugging it back in (3.29), we get

$$n_{\text{iter}}(\epsilon) \leq \left\lceil d_{\max} \frac{|\log \epsilon|}{\lambda_{\min}(L)} \right\rceil,$$

which proves the upper bound on the number of iterations required. The proof of the lower bound is similar, and is therefore omitted. ■

# Chapter 4

## Optimal estimation in infinite graphs and electrical analogy

### 4.1 Introduction

In this chapter we take the first steps, and develop the tools needed, toward answering the error scaling question raised in Chapter 1. As discussed in Chapter 2, we examine the error scaling of the optimal estimator (BLUE), since it achieves the minimum variance among all linear unbiased estimators, and therefore gives us an algorithm independent limit on estimation accuracy.

We focus our attention on large measurement graphs in answering the error scaling question. A large measurement graph can result from a sensor-actuator network obtained by deploying a large number of nodes. Sensor networks consisting of thousands of wireless nodes are already in test and deployment phase [13]. Large networks are being envisioned for civil as well as defense applications [101]. Large measurement graphs can also result with a small number of physical agents

that are mobile, since in that case the measurement graph consists of all the variables and measurements that appear over a time interval of interest. For example, in the multi-robot localization example described in Section 2.1.4, the nodes of the measurement graph are the positions of the mobile robots at various time instants. Upon collecting all the relative measurements over a time interval  $(0, t)$ , we obtain a measurement graph  $\mathcal{G}(t)$ . In an experiment described in [102], a team of 80 robots were deployed in a “search and protect” mission. If every robot takes relative position measurements every minute with three other robots on an average, after an hour the measurement graph will consist of 4800 nodes and 14400 edges. Thus, large measurement graphs are quite likely in sensor-actuator networks.

To investigate error scaling in a large graph, we consider the limiting situation of an infinite graph, with a countable number of nodes and edges (i.e., variables and relative measurements). The results in this Chapter show that as one considers larger and larger numbers of measurements, the minimum estimation error covariance of a node variable tends to a limiting covariance matrix that is positive definite. This limiting covariance is characterized by the effective resistance in an abstract electrical network in which currents, voltages and resistances are matrix-valued. The main assumption needed is that the graph must have bounded degree.

It is often easier to establish asymptotic results for infinite graphs than for large finite graphs, since boundary effects are usually weaker in infinite graphs than in finite graphs. This advantage is exploited in the next Chapter in obtaining error scaling laws in infinite measurement graphs. The results of this chapter provides a formal justification for regarding infinite graphs as suitable proxies for large but finite graphs.

*Chapter Organization:* After stating the problem addressed in this Chapter

precisely in Section 4.2, we summaries the main results and related prior work in Section 4.3. Generalized electrical networks are introduced in Section 4.4, along with a few technical results on such networks, so that the main results of the chapter can be stated, which is done in Section 4.5. We go back to discuss in more detail generalized electrical networks in Section 4.6, which also describes a few technical, but crucial results needed to establish the main results of this Chapter (and also of the next). Section 4.7 provides the proof of the main results, which is followed by a discussion on relevant open problems in Section 4.8.

## 4.2 Problem statement

Consider an *infinite* measurement graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  with a single reference node  $o \in \mathcal{V}$ , where the node set  $\mathcal{V}$  and edge-set  $\mathcal{E}$  are countably infinite. The measurement error covariances are specified by an edge-covariance function  $P : \mathcal{E} \rightarrow \mathbb{S}^{k+}$ , where  $\mathbb{S}^{k+}$  is the set of  $k \times k$  symmetric positive definite matrices. The pair  $(\mathcal{G}, P)$  is called a *measurement network*. Imagine, for the moment, that we are interested in the estimate of a particular node  $u \in \mathcal{V}$ . Let  $\mathcal{G}_{\text{finite}} = (\mathcal{V}_{\text{finite}}, \mathcal{E}_{\text{finite}})$  be a finite subgraph of the infinite graph  $\mathcal{G}$ , that contains both  $u$  and  $o$ , i.e.,  $\mathcal{V}_{\text{finite}} \subset \mathcal{V}$ , and  $\mathcal{E}_{\text{finite}} \subset \mathcal{E}$ . One can regard the subgraph  $\mathcal{G}_{\text{finite}}$  as consisting of those measurements that are processed upto some time  $t < \infty$ . Given the finite measurement network  $(\mathcal{G}_{\text{finite}}, P)$ , it is straightforward to compute the optimal estimate  $\hat{x}_u^*(\mathcal{G}_{\text{finite}})$  of the unknown node variable  $x_u$  in the network  $(\mathcal{G}_{\text{finite}}, P)$ , as described in Chapter 2. The covariance matrix of the error in this estimate is

$$\Sigma_{u,o}(\mathcal{G}_{\text{finite}}) := \mathbb{E}[(x_u - \hat{x}_u^*(\mathcal{G}_{\text{finite}}))(x_u - \hat{x}_u^*(\mathcal{G}_{\text{finite}}))^T],$$

which exists as long  $\mathcal{G}_{\text{finite}}$  is weakly connected (see Theorem 2.2.1). As time goes by, one can process larger subsets of measurements, which can be visualized

by a sequence of progressively larger finite subgraphs of  $\mathcal{G}$ . In this context, we are interested in studying if there is a limit to the estimation accuracy achievable by processing more and more measurements. Specifically, we want to know if there is a point beyond which there is little gain in processing more measurements, as it will not improve the estimate of  $x_u$  significantly. This raised the following question:

**Limiting accuracy in infinite graphs:** *Consider an infinite measurement network  $(\mathcal{G}, P)$ , with the graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  that has a single reference node  $o \in \mathcal{V}$ . For every node variable  $x_u, u \in \mathcal{V}$ , what is the minimum possible estimation error covariance that can be achieved by using an arbitrarily large, finite subset of the measurements available in  $\mathcal{E}$ ?*

Once we characterize the minimum possible estimation error of a node variable in infinite graphs, in Chapter 5 we examine how this error varies as a function of the node's distance from the reference, and how this variation depends on the structure of the graph. An answer to the error scaling question raised in Chapter 1 is thus answered.

A measurement network is assumed to satisfy the following assumption.

**Assumption 4.2.1 (measurement network).** The measurement network  $(\mathcal{G}, P)$  satisfies the following properties:

1. The graph  $\mathcal{G}$  is weakly connected.
2. The graph  $\mathcal{G}$  has a finite maximum node degree<sup>1</sup>.
3. The edge-covariance function  $P$  is uniformly bounded, i.e., there exists constant symmetric positive matrices  $P_{\min}, P_{\max}$  such that  $P_{\min}I_k \leq P_e \leq P_{\max}I_k, \forall e \in \mathcal{E}$ .

---

<sup>1</sup>The degree of a node is the number of edges that are incident on the node. An edge  $(u, v)$  is said to be incident on the nodes  $u$  and  $v$ .

4. The measurement errors on distinct edges are uncorrelated, i.e., if  $e$  and  $e'$  are two distinct edges in  $\mathcal{G}$ , then  $E[\epsilon_e \epsilon_{e'}^T] = 0$ .  $\square$

In the above, for two matrices  $A, B \in \mathbb{R}^{k \times k}$ ,  $A > (\geq) B$  means  $A - B$  is positive definite (semidefinite), which is also written as  $A - B > (\geq) 0$ . We write  $A < (\leq) B$  if  $-A > (\geq) -B$ .

To formulate the problem of determining the limiting accuracy in infinite graphs precisely, we define the *limiting BLUE error covariance*  $\Sigma_{u,o}$  for a node variable  $x_u$  in an infinite measurement network  $(\mathcal{G}, P)$  as

$$\Sigma_{u,o} = \inf_{\mathcal{G}_{\text{finite}}} \Sigma_{u,o}(\mathcal{G}_{\text{finite}}), \quad (4.1)$$

where the infimum is taken over all finite subgraphs  $\mathcal{G}_{\text{finite}}$  of  $\mathcal{G}$  that contain the nodes  $u$  and  $o$ . We define a matrix  $M \in \mathbb{S}^{k+}$  to be the *infimum* of the set  $S$ , where  $S \subset \mathbb{S}^{k+}$ , and write

$$M = \inf S, \quad (4.2)$$

if  $M \leq A$  for every matrix  $A \in S$ , and for every positive real  $\epsilon$ , there exists  $B \in S$  such that  $M + \epsilon I_k > B$ . We will show in Section 4.5 that under Assumption 4.2.1, the infimum  $\Sigma_{u,o}$  in (4.1) is well-defined and is a symmetric positive definite matrix. In the sequel, we will often say “limiting BLUE covariance in a network” without specifying if the network is finite or infinite. When the network is finite, this phrase should be understood to mean simply the BLUE covariance.

Note that the BLUE covariance above is *not defined* in terms of the error in the estimate obtained by using all the infinite number of measurements available in  $\mathcal{G}$ . There are two reasons for it. First, in practice, one may have a very large number of measurements but never an infinite number of them. So establishing the limit of estimation accuracy that is achievable by using arbitrarily large but

a finite number of measurements is more relevant from a practical point of view. The second reason is that characterizing the best linear unbiased estimator and its covariance for an infinite number of measurements is technically challenging.

### 4.3 Contributions and prior work

The results established in this chapter are the following:

1. In a measurement network (finite or infinite), the limiting BLUE error covariance  $\Sigma_{u,o}$  that can be achieved by using arbitrarily large finite subsets of measurements is equal to a matrix-valued effective resistance  $R_{u,v}^{\text{eff}}$  between  $u$  and  $o$  in a generalized electrical network, in which currents, voltages and resistances are matrix valued. The electrical network is obtained by assigning to every edge of the measurement graph a matrix-valued resistance equal to the covariance of the measurement error on that edge. This result is called the *electrical analogy*.
2. We show that for every positive constant  $\epsilon > 0$ , it is possible to construct an unbiased estimate for a node variable  $x_u$  using only a finite subset of the available measurements but whose estimation error covariance is only  $\epsilon$  above the minimum possible estimation error variance that could be obtained by considering an arbitrarily large number of the available measurements.

This convergence result provides the formal justification for regarding infinite graphs as suitable proxies for very large but finite graphs, and establishes the conditions under which such approximation is valid. In particular, the assumption of finite maximum node degree is needed.

Another implication of this result is that for estimation problems based on relative measurements, after a certain point, considering more measurements will only marginally improve the quality of the estimate. On the positive side, this simplifies the construction of estimation algorithms in large-scale networks because it justifies considering a relatively small subset of measurements. Although in Chapter 3, distributed algorithms were proposed and analyzed to compute the optimal estimates, for a large network these algorithms may take a long time to provide accurate estimates. The reason is that information about all the available measurements are fused iteratively to determine the estimate of every node variable. The results of this chapter suggest that it may be possible to devise algorithms such that they obtain estimates quite fast, while sacrificing little accuracy.

3. The BLUE covariance of a node variable can only decrease upon using more measurements, and can only increase upon using fewer measurements. This monotonicity of BLUE covariances is a result of our extension of a classical result for electrical networks, named Rayleigh’s monotonicity law [2], to generalized electrical networks. This monotonicity property of matrix-valued effective resistances is crucial not only in proving all the results in this chapter, but also in establishing the error scaling laws in Chapter 5.

*Prior work:* to the best of our knowledge, the question on limiting accuracy in infinite graphs posed and answered in this chapter has not been investigated earlier. The analogy between estimator variance and effective resistance, when the node variables are scalars, was noted by Karp *et al.* [73] in connection with the time-synchronization problem. Here we show that an electrical analogy still holds for vector-valued node variables, provided that we consider *generalized electrical networks* in which currents, voltages, and resistors are matrix-valued. Another

important distinction with [73] is that the measurement graph was finite in Karp *et al.* [73], whereas we allow the measurement graph to be infinite.

The extension of the electrical analogy to infinite graphs requires, among other things, that the currents and voltages in an infinite, generalized electrical network are well defined. Our proof that this is so parallels the work of Flanders, who in 1971 provided perhaps the earliest exposition of infinite electrical networks [103]. Although our proof technique is different, the results for infinite generalized electrical networks are direct extensions of Flanders' results for infinite scalar electrical networks.

## 4.4 Generalized electrical networks

A *generalized electrical network*  $(\mathcal{G}, R)$  consists of a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  (finite or infinite) together with a function  $R : \mathcal{E} \rightarrow \mathbb{S}^{k+}$  that assigns to each edge  $e \in \mathcal{E}$  a symmetric positive definite matrix  $R_e$  called the *generalized resistance* of the edge.

Recall that a *generalized flow* from node  $u \in \mathcal{V}$  to node  $v \in \mathcal{V}$  with intensity  $\mathbf{j} \in \mathbb{R}^{k \times k}$  is an edge-function  $j : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  such that

$$\sum_{\substack{(p,q) \in \mathcal{E} \\ p = \bar{p}}} j_{p,q} - \sum_{\substack{(q,p) \in \mathcal{E} \\ p = \bar{p}}} j_{q,p} = \begin{cases} \mathbf{j} & \bar{p} = u \\ -\mathbf{j} & \bar{p} = v \\ \mathbf{0} & \text{otherwise} \end{cases} \quad \forall \bar{p} \in \mathcal{V}. \quad (4.3)$$

We say that a flow  $i$  is a *generalized current* when there is a *node-function*  $V : \mathcal{V} \rightarrow \mathbb{R}^{k \times k}$  for which

$$R_{u,v} i_{u,v} = V_u - V_v, \quad \forall (u, v) \in \mathcal{E}. \quad (4.4)$$

The node-function  $V$  is called a *generalized potential associated with the current  $i$* . Eq. (4.3) should be viewed as a generalized version of Kirchoff's current law and can be interpreted as: the net flow out of each node other than  $u$  and  $v$  is equal to zero, whereas the net flow out of  $u$  is equal to the net flow into  $v$  and both are equal to the flow intensity  $\mathbf{j}$ . Eq. (4.4) provides in a combined manner, a generalized version of Kirchoff's loop law, which states that the net potential drop along a circuit must be zero, and Ohm's law, which states that the potential drop across an edge must be equal to the product of its resistance and the current flowing through it. A circuit is an undirected path that start and end at the same node. For  $k = 1$ , generalized electrical networks are the usual electrical networks with scalar currents, potentials, and resistors.

The *energy dissipated* by an edge-function  $j$  in the network  $(\mathcal{G}, R)$  is defined by

$$\|j\| := \left( \sum_{e \in \mathcal{E}} \text{trace}(j_e^T R_e j_e) \right)^{\frac{1}{2}}. \quad (4.5)$$

It is straightforward to verify that the set of edge-functions with finite dissipated energy constitutes a Hilbert space  $\mathcal{H}_R$  with inner product  $\langle j, \bar{j} \rangle = \sum_{e \in \mathcal{E}} \text{trace}(j_e^T R_e \bar{j}_e)$ ,  $\forall j, \bar{j} \in \mathcal{H}_R$ . For infinite networks, the summation in (4.5) is an absolutely convergent series and the order of summation is irrelevant. Flows of finite support always belong to  $\mathcal{H}_R$ .

#### 4.4.1 Existence and uniqueness of generalized current

Existence and uniqueness of scalar currents in infinite networks has been examined in [103, 104]. It was shown by Flanders that, unlike in finite networks, in an infinite electrical network the current is not uniquely determined by Kirchoff's laws and Ohm's law [103]. He showed, however, that uniqueness of current in

an infinite network can be established if two additional conditions are imposed: the current has a finite dissipated energy and it is the limit of flows with finite support. For this reason, in examining the uniqueness of generalized currents in infinite networks we restrict ourselves to generalized flows that are limits of finite support flows and that have finite dissipated energy. For finite networks these conditions hold trivially.

The following theorem establishes existence, uniqueness, and linearity of generalized currents and potential differences in generalized electric networks. The proof of this result is provided in Section 4.9.

**Theorem 4.4.1 (Generalized Current).** *In a generalized electrical network  $(\mathcal{G}, R)$  that satisfies Assumption 4.2.1, for every pair of nodes  $u, v \in \mathcal{V}$  and for every intensity  $\mathbf{i} \in \mathbb{R}^{k \times k}$ , among all flows that have finite dissipated energy and are limits of finite support flows, there exists a unique current  $i$  from  $u$  to  $v$  with intensity  $\mathbf{i}$ . In addition,*

1. *the current is the flow that minimizes the energy dissipation, among all flows from node  $u$  to node  $v$  with intensity  $\mathbf{i}$ , that are limits of finite support flows, and*
2. *the current  $i$  and the potential difference  $V_p - V_q$  (for every  $p, q \in \mathcal{V}$ ) are linear functions of the intensity  $\mathbf{i}$ . The potential is unique only upto an additive constant.* □

It was previously known that in a scalar electrical network, the current minimizes energy dissipation. This result is known as Thomson's Minimum Energy Principle [2, 104]. Theorem 4.4.1 shows that generalized currents also obey Thomson's Principle in both finite and infinite networks.

### 4.4.2 Generalized effective resistance

It was shown in the previous section that the potential difference  $V_u - V_v \in \mathbb{R}^{k \times k}$  associated with a current of intensity  $\mathbf{i} \in \mathbb{R}^{k \times k}$  flowing from  $u$  to  $v$  is a linear function of  $\mathbf{i}$ . It turns out that this linear map can be expressed through the matrix multiplication by a  $k \times k$  matrix, which is stated next. The proof of this result is provided in Section 4.9.

**Lemma 4.4.1.** *Let  $(\mathcal{G}, R)$  be a generalized electric network satisfying Assumption 4.2.1. The linear mapping between  $\mathbf{i}$  and  $V_u - V_v$  can be defined by multiplication by a  $k \times k$  matrix, which we call the generalized effective resistance  $R_{u,v}^{\text{eff}}$  between  $u$  and  $v$ :*

$$V_u - V_v = R_{u,v}^{\text{eff}} \mathbf{i}, \quad \forall \mathbf{i} \in \mathbb{R}^{k \times k}. \quad \square$$

In the sequel, we will refer to generalized effective resistance simply as effective resistance. In view of Lemma 4.4.1, the effective resistance between two nodes is the potential difference between them when a current with intensity  $I_k$ , the  $k \times k$  identity matrix, is injected at one node and extracted at the other, which is analogous to the definition of effective resistance in scalar networks [2]. Moreover, the effective resistance is a symmetric positive-definite matrix. To show this, we will need the following technical result (also proved in Section 4.9), which will have additional usefulness in the sequel.

**Lemma 4.4.2.** *Let  $i \in \mathcal{H}_R$  be the unique current in the network  $(\mathcal{G}, R)$  with intensity  $\mathbf{i} \in \mathbb{R}^{k \times k}$  from  $u$  to  $v$ , and let  $j$  be a flow with intensity  $\mathbf{j} \in \mathbb{R}^{k \times k}$  from  $u$  to  $v$  that can be expressed as a limit of finite support flows. Then,*

$$\sum_{e \in \mathcal{E}} i_e^T R_e j_e = (V_u - V_v)^T \mathbf{j},$$

where  $V$  is a generalized potential associated with the current  $i$ . Moreover, the series in the left-hand side converges absolutely, meaning that each one of the  $k^2$  series that constitute the matrix-valued left hand side converges absolutely.  $\square$

To prove positive-definiteness of effective resistances, set  $j = i$  in Lemma 4.4.2, where both  $i$  and  $j$  have intensity  $I_k$ , to obtain

$$\sum_{e \in \mathcal{E}} i_e^T R_e i_e = (V_u - V_v)^T = (R_{u,v}^{\text{eff}})^T, \quad (4.6)$$

where the second equality follows from the definition of effective resistance in Lemma 4.4.1. Since all the generalized edge-resistances  $R_e$  are symmetric and positive-definite, we conclude that the left-hand side must be symmetric and positive-definite, which confirms that effective resistances are indeed symmetric positive-definite. This is stated below for future reference:

**Proposition 4.4.1.** *For every pair of nodes  $u, v$  in a generalized electrical network  $(\mathcal{G}, P)$  that satisfies Assumption 4.2.1, the generalized effective resistance between them  $R_{u,v}^{\text{eff}}$  is a symmetric positive definite matrix.*  $\square$

## 4.5 Main results

Given a measurement network  $(\mathcal{G}, P)$ , we construct an analogous generalized electrical network  $(\mathcal{G}, P)$ , that is, by assigning to every edge a matrix-valued resistance equal to the measurement error covariance of that edge. The generalized effective resistances in the electrical network  $(\mathcal{G}, P)$  precisely characterizes the minimum estimation error covariances achievable in the measurement network  $(\mathcal{G}, P)$ , which is stated in the next theorem. The proof of the theorem is provided in Section 4.7.

**Theorem 4.5.1 (Electrical Analogy).** *Consider a measurement network  $(\mathcal{G}, P)$  satisfying Assumption 4.2.1 with  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  and a single reference node  $o \in \mathcal{V}$ . Then, for every node  $u \in \mathcal{V} \setminus \{o\}$ , the limiting BLUE error covariance  $\Sigma_{u,o}$  is given by*

$$\Sigma_{u,o} = R_{u,o}^{\text{eff}},$$

where  $R_{u,o}^{\text{eff}}$  is the effective resistance between  $u$  and  $o$  in the generalized electrical network  $(\mathcal{G}, P)$ .  $\square$

The proof of this result will follow directly from a more general result that also explains what happens to the estimates constructed by using arbitrarily large, finite subsets of the available measurements. To state the result, we need some preliminaries. For two graphs  $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ ,  $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ , the notation  $\mathcal{G}_1 \subset \mathcal{G}_2$  means  $\mathcal{V}_1 \subset \mathcal{V}_2$  and  $\mathcal{E}_1 \subset \mathcal{E}_2$ . We now consider a sequence of finite measurement subgraphs  $\{\mathcal{G}_{(n)} = (\mathcal{V}_{(n)}, \mathcal{E}_{(n)})\}$  that satisfies the following assumption.

**Assumption 4.5.1 (Nested Sequence).** A sequence of finite graphs  $\mathcal{G}_{(1)}, \mathcal{G}_{(2)}, \mathcal{G}_{(3)}, \dots$  has the following properties:

1. The sequence is *nested* in the sense that

$$\mathcal{G}_{(1)} \subset \mathcal{G}_{(2)} \subset \mathcal{G}_{(3)} \subset \dots \subset \mathcal{G},$$

2. The sequence *converges* to the graph  $\mathcal{G}$  in the sense that every node and edge in  $\mathcal{G}$  appears in one of the  $\mathcal{G}_{(n)}$  for some finite  $n$ .
3. Each finite graph  $\mathcal{G}_{(n)}$ ,  $n \in \mathbb{N}$  is weakly connected.  $\square$

When investigating the error in  $x_u$ 's estimate, every graph  $\mathcal{G}_{(n)}$  in such a nested sequence of finite graphs should contain the reference node  $o$  and the node

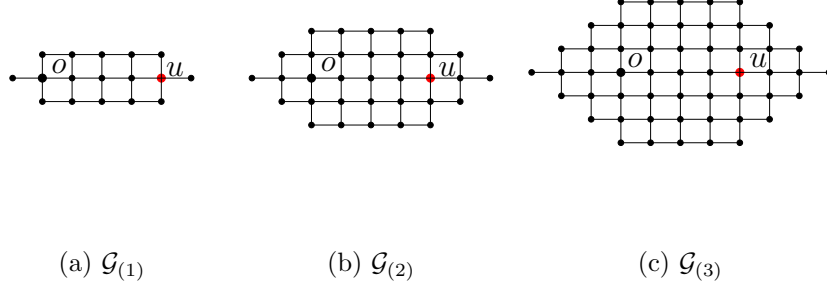


Figure 4.1. A nested sequence of measurement graphs that “tend to” the 2-dimensional square lattice.

of interest  $u$ . Figure 4.1 shows the first few elements of such a nested graph sequence that will eventually converge to the 2-dimensional square lattice (the formal definition of a lattice will be provided in Section 5.5.1). One could regard each finite subgraph  $\mathcal{G}_{(n)}$  as describing a finite subset of available measurements that could be processed up to some time  $t_n < \infty$  to construct an estimate of  $x_u$ . As time increases, more measurements can be processed, and therefore at some time  $t_{n+1} > t_n$ , the subgraph  $\mathcal{G}_{(n+1)}$  contains more measurements than  $\mathcal{G}_{(n)}$ . In this context, we are interested in studying if there is a point after which there is little improvement in estimation error upon processing more measurements, and whether or not the sequence of estimates produced using the nested sequence of subgraphs converges.

Let  $\hat{x}_u^{(n)}$  denote the best linear unbiased (BLU) estimate of  $x_u$  in the finite measurement network  $(\mathcal{G}_{(n)}, P)$ ; Chapter 2 describes how to compute this estimate. This estimate is a linear combination of the measurements  $\zeta_e, e \in \mathcal{E}$  specified by a set of appropriately chosen coefficient matrices. In particular, the BLU estimate

is given by

$$\hat{x}_u^{(n)} = \sum_{e \in \mathcal{E}_{(n)}} C_e^{(n)T} \zeta_e, \quad (4.7)$$

where the function  $C^{(n)} : \mathcal{E}_{(n)} \rightarrow \mathbb{R}^{k \times k}$  specifies the coefficients of the measurements. Note that in the equation above, and in the sequel, for a function  $f$  with the edge set  $\mathcal{E}$  as the domain, we use  $f_e$  to denote the value of the function at an edge  $e \in \mathcal{E}$ . We call the function  $C^{(n)}$  the *BLU estimator for  $x_u$  based on the finite graph  $\mathcal{G}_{(n)}$* .

Every estimator  $C^{(n)}$  can be viewed as an element of the real linear vector space  $\mathcal{H}_P$  consisting of all *edge-functions* of the form  $C : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  for which

$$\|C\|^2 := \sum_{e \in \mathcal{E}} \text{trace}(C_e^T P_e C_e) < \infty, \quad (4.8)$$

where each  $P_e$  denotes the error covariance matrix for the measurement associated with the edge  $e \in \mathcal{E}$ . It is straightforward to show that  $\mathcal{H}_P$  is a Hilbert space with the associated inner product  $\langle C, \bar{C} \rangle = \sum_{e \in \mathcal{E}} \text{trace}(C_e^T P_e \bar{C}_e)$ ,  $\forall C, \bar{C} \in \mathcal{H}_P$ . We say that an edge-function in  $\mathcal{H}_P$  has finite support if it has only a finite number of nonzero entries. Since all the sets  $\mathcal{E}_{(n)}$  in (4.7) are finite, every estimator  $C^{(n)}$  is a finite-support edge-function in  $\mathcal{H}_P$ .

For infinite graphs, the summation in (4.8) is actually a series. However, the series is absolutely convergent due to the positive definiteness of the  $P_e$ 's, hence the order of the summation is immaterial and therefore the expression in (4.8) is well defined.

We now state the second main result of the chapter, which establishes the convergence of BLU estimators and of the estimates, as  $n \rightarrow \infty$ . In the statement of the theorem,  $x_u^{(n)}$  denotes the BLU estimate of  $x_u$  in the finite graph  $\mathcal{G}_{(n)}$ : The proof is provided in Section 4.7.

**Theorem 4.5.2 (BLUE Convergence).** *Consider a network  $(\mathcal{G}, P)$  with a single reference node  $o \in \mathcal{V}$  that satisfies Assumption 4.2.1. For every node  $u \in \mathcal{V} \setminus \{o\}$ , if  $\{\mathcal{G}_{(n)}\}$  is a nested sequence of finite graphs that satisfies Assumption 4.5.1 with  $u$  and  $o$  belonging to every graph in the sequence, the following statements hold.*

1. *The sequence of BLU estimates  $\{\hat{x}_u^{(n)}\}$  converges in the mean-square sense.*
2. *The sequence of BLU estimators  $\{C^{(n)}\}$  for  $x_u$  converges to some  $C \in \mathcal{H}_P$ .*
3. *The sequence of BLUE estimation error covariance matrices*

$$\Sigma_{u,o}^{(n)} := \mathbb{E}[(x_u - \hat{x}_u^{(n)})(x_u - \hat{x}_u^{(n)})^T]$$

*converges to the effective resistance  $R_{u,o}^{\text{eff}}$  between  $u$  and  $o$  in the electrical network  $(\mathcal{G}, P)$ , i.e.,*

$$\lim_{n \rightarrow \infty} \Sigma_{u,o}^{(n)} = R_{u,o}^{\text{eff}}.$$

*Moreover, the BLUE covariances decrease monotonically, in the sense that*

$$\Sigma_{u,o}^{(1)} \geq \Sigma_{u,o}^{(2)} \geq \dots \quad \square$$

Theorem 4.5.2 shows that under the bounded degree assumption, by using only a finite number of measurements among the infinitely many potentially available, we can construct estimates whose error variance is arbitrarily close to the minimum possible variance that could be achieved by using an arbitrarily large number of measurements. In addition, the estimates themselves converge and the “limiting” estimator is square-summable in the sense of (4.8). The theorem tells us that even when the number of measurements go to infinity, the limiting BLUE covariance does not go to 0, but to a positive definite matrix.

Proofs of these results will require developing additional tools by exploiting the analogy between generalized electrical networks and measurement networks, which is done in Section 4.6. We briefly discuss how fast the convergence of  $\Sigma_{u,o}^{(n)}$  to  $R_{u,o}^{\text{eff}}$  takes place.

### 4.5.1 Convergence rate

Theorem 4.5.2 shows that the BLU estimator error variance in a sequence of nested finite subgraphs of an infinite measurement graph converges to a limiting variance that is numerically equal to an effective resistance, regardless of how the sequence  $\mathcal{G}_{(n)}$  is constructed. However, the rate at which the covariances  $\Sigma_{u,o}^{(n)}$  converge to the effective resistance in the infinite graph will depend on how the sequence  $\{\mathcal{G}_{(n)}\}$  is constructed vis-a-vis the nodes  $u$  and  $o$ . One natural way to construct the graph  $\mathcal{G}_{(n)} = (\mathcal{V}_{(n)}, \mathcal{E}_{(n)})$  is to take  $\mathcal{V}_{(n)}$  to contain all nodes that are at a graphical distance smaller than  $\alpha(n)$  from the shortest path connecting  $u$  and  $o$ , where  $\alpha(\cdot)$  is a positive and increasing function. The distance of a node from a path denotes the minimum graphical distance between the node and any node lying on the path. If there are multiple shortest paths, we take the union of the sets obtained for each of the shortest paths.  $\mathcal{E}_{(n)}$  is then chosen as the set of edges that are incident on the nodes in  $\mathcal{V}_{(n)}$ . This construction satisfies Assumption 4.5.1.

Figure 4.2(a-c) shows the first three members of a sequence of nested subgraphs  $\{\mathbf{Z}_{2(n)}\}$  of the 2-dimensional lattice  $\mathbf{Z}_2$ , constructed according to the procedure outlined above, with  $\alpha(n) = n$ . For simplicity, we consider the case of scalar variables and measurements, and every measurement error is assumed to have a variance 1. Covariances for vector-valued variables ( $k > 1$ ) could be obtained

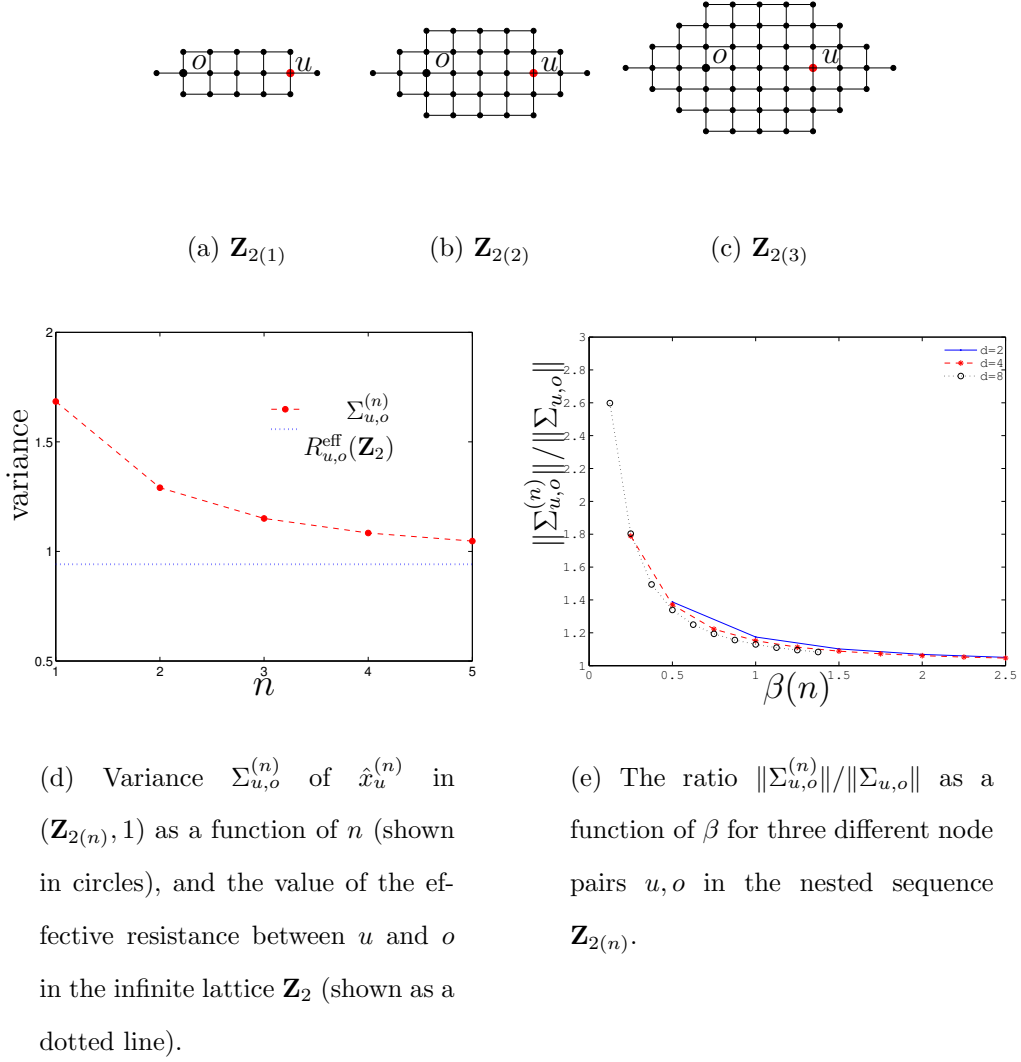


Figure 4.2. (a)-(c) The first three members of a sequence of nested subgraphs  $\mathbf{Z}_{2(n)}$  of the 2-dimensional lattice  $\mathbf{Z}_2$ , and (d) the plot of variances  $\Sigma_{u,o}^{(n)}$  in the sequence of measurement networks  $(\mathbf{Z}_{2(n)}, 1)$  as a function of  $n$ . (e) Trend of the ratio of variance in the finite subnetworks  $(\mathbf{Z}_{2(n)}, 1)$  to the minimum possible variance in  $(\mathbf{Z}_2, 1)$ , as a function of  $\beta(n)$  for three different node pairs  $u, o$ .

using Lemma 4.6.1. Figure 4.2(d) shows the plot of the variances  $\Sigma_{u,o}^{(n)}$  of node  $u$  in the measurement network  $(\mathbf{Z}_{2(n)}, 1)$  as a function of  $n$ . The limiting value

of the variance is the effective resistance between  $u$  and  $o$  in the infinite lattice  $\mathbf{Z}_2$ . In an infinite 2-dimensional lattice with unit resistance on every edge, the effective resistance between two nodes  $u$  with relative  $x$  and  $y$  coordinates is given by  $R_{u,o}^{\text{eff}} = \frac{1}{\pi}(\log \sqrt{x^2 + y^2} + \gamma + \frac{1}{2} \log 8)$ , where  $\gamma \approx 0.577$  [105]. For the example in Figure 4.2(a-c),  $x = 4, y = 0$ , so the limiting variance for node  $u$  is  $\Sigma_{u,o} = R_{u,o}^{\text{eff}} \approx 0.956$ , which is shown by a dotted line in the Figure 4.2(d). As expected, the variances  $\Sigma_{u,o}^{(n)}$  monotonically decrease and approach the asymptotic value as  $n$  increases.

For a given nested sequence  $\mathcal{G}_{(n)}$ , the convergence rate of  $\Sigma^{(n)}$  to  $\Sigma$  will depend on the graphical distance  $d_{u,o}$  between nodes  $u$  and  $o$ . Taking this into account, we can construct the sequence  $\mathcal{G}_{(n)}$  by choosing  $\mathcal{V}_{(n)}$  as the set of nodes that are within a graphical distance of  $\beta(n)d_{u,o}$  of the shortest path connecting  $u$  and  $o$ , where  $\beta(\cdot)$  is a positive and increasing function. Numerical studies on the 2-dimensional lattice  $\mathbf{Z}_2$  indicate that with this construction, the ratio  $\|\Sigma_{u,o}^{(n)}\|/\|\Sigma_{u,o}\|$  depends only on the value of  $\beta$  and is independent of  $d_{u,o}$ . Figure 4.2(e) shows the ratio  $\|\Sigma_{u,o}^{(n)}\|/\|\Sigma_{u,o}\|$  as a function of  $\beta$  for three different nodes taken at distances of 2, 4 and 8, respectively, from  $o$ . The figure shows that the rate of convergence of  $\Sigma_{u,o}^{(n)}$  to the limiting value  $\Sigma_{u,o}$  is not sensitive to the distance between  $u$  and  $o$ . In particular, with  $\beta = 2$ , the error between  $\Sigma_{u,o}^{(n)}$  and  $\Sigma_{u,o}$  is less than 10%.

These studies show that in a 2-dimensional lattice, a relatively small subgraph is sufficient to obtain an estimate whose variance is quite close to the minimum possible achievable by using all the measurements. For an arbitrary measurement graph, as long as the graph is “close to” a lattice in an appropriate sense, similar trends are expected. Appropriate measures of closeness to lattices are developed in Chapter 5.

## 4.6 Properties of generalized electrical networks

### 4.6.1 Rayleigh's monotonicity law

The next result relates the effective resistances of two distinct networks related by an appropriate partial order. A similar result for finite scalar networks, called Rayleigh's Monotonicity Law [2], states that if the edge-resistances in a scalar electrical network are increased (perhaps even made infinity, i.e., an open circuit), then the effective resistance between every pair of nodes in the network can only increase. For a long time, Rayleigh's Monotonicity Law was considered so evidently true that no proof was deemed necessary. Nevertheless, Doyle and Snell [2] provided a proof, which we now extend to generalized electrical networks.

To present the result in full generality, we introduce the notion of graph embedding. We say that a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  can be *embedded* in another graph  $\bar{\mathcal{G}} = (\bar{\mathcal{V}}, \bar{\mathcal{E}})$  if  $\mathcal{V} \subset \bar{\mathcal{V}}$ , and, for every edge between two nodes in  $\mathcal{G}$ , there is a corresponding edge between them in  $\bar{\mathcal{G}}$ . More precisely,  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$  if

1. there exists an injective map  $\eta : \mathcal{V} \rightarrow \bar{\mathcal{V}}$ , and
2. for every  $e \in \mathcal{E}$ , there exists  $\bar{e} \in \bar{\mathcal{E}}$  such that, if  $e \sim u, v$  then  $\bar{e} \sim \eta(u), \eta(v)$ .

In other words,  $\mathcal{G}$  is a subgraph of  $\bar{\mathcal{G}}$  when they are thought of as undirected. In the sequel, we use  $\mathcal{G} \subset \bar{\mathcal{G}}$  to denote that  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$ . In addition, when  $\eta : \mathcal{V} \rightarrow \bar{\mathcal{V}}$  is the embedding of  $\mathcal{G}$  into  $\bar{\mathcal{G}}$ , for every edge  $e \in \mathcal{E}$ , we use the somewhat loose notation  $\eta(e)$  to denote the edge in  $\bar{\mathcal{E}}$  that corresponds to the edge  $e$ .

**Theorem 4.6.1 (Generalized Rayleigh’s Monotonicity Law).** *Consider two generalized electrical networks  $(\mathcal{G}, R)$  and  $(\bar{\mathcal{G}}, \bar{R})$  for which  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$  with an embedding function  $\eta : \mathcal{V} \rightarrow \bar{\mathcal{V}}$ , i.e.,  $\mathcal{G} \subset \bar{\mathcal{G}}$ , and  $R_e \geq \bar{R}_{\eta(e)}$  for every  $e \in \mathcal{E}$ . For every pair of nodes  $u, v$  of  $\mathcal{G}$ ,*

$$R_{u,v}^{\text{eff}} \geq \bar{R}_{u,v}^{\text{eff}},$$

*where  $R_{u,v}^{\text{eff}}$  and  $\bar{R}_{u,v}^{\text{eff}}$  are the effective resistances between  $u$  and  $v$  in the networks  $(\mathcal{G}, R)$  and  $(\bar{\mathcal{G}}, \bar{R})$ , respectively.  $\square$*

*Proof of Theorem 4.6.1.* Let  $i : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  and  $\bar{i} : \bar{\mathcal{E}} \rightarrow \mathbb{R}^{k \times k}$  be the currents from  $u$  to  $v$  in the networks  $(\mathcal{G}, R)$  and  $(\bar{\mathcal{G}}, \bar{R})$ , respectively, both with intensity  $\mathbf{i} \in \mathbb{R}^{k \times k}$ . Denote by  $\eta(\mathcal{E})$  the set of edges in  $\bar{\mathcal{E}}$  that correspond to the edges in  $\mathcal{E}$ . Define  $\bar{j} : \bar{\mathcal{E}} \rightarrow \mathbb{R}^{k \times k}$  to be the following “extension” of the current  $i$  to the graph  $\bar{G}$

$$\bar{j}_e = \begin{cases} i_{\eta^{-1}(e)} & e \in \eta(\mathcal{E}) \\ 0 & e \in \bar{\mathcal{E}} \setminus \eta(\mathcal{E}) \end{cases}$$

where  $\eta^{-1}(e)$  represents the pre-image of  $e$  in the set  $\mathcal{E}$ , thinking of  $\eta$  as a mapping from  $\mathcal{E}$  to  $\bar{\mathcal{E}}$ . We conclude that  $\bar{j}$  satisfies Kirchoff’s current law (4.3) for the network  $(\bar{\mathcal{G}}, \bar{R})$  and is therefore a flow for this network (although not necessarily a current). Since according to Theorem 4.4.1 the current  $\bar{i}$  is the flow of minimum

dissipated energy for the network  $(\bar{\mathcal{G}}, R)$ , we conclude that

$$\begin{aligned}
\text{trace}\left(\sum_{e \in \bar{\mathcal{E}}} \bar{i}_e^T \bar{R}_e \bar{i}_e\right) &\leq \text{trace}\left(\sum_{e \in \bar{\mathcal{E}}} \bar{j}_e^T \bar{R}_e \bar{j}_e\right) \\
&\leq \text{trace}\left(\sum_{e \in \eta(\mathcal{E})} \bar{j}_e^T \bar{R}_e \bar{j}_e\right) \\
&= \text{trace}\left(\sum_{e \in \mathcal{E}} i_e^T \bar{R}_e i_e\right) \\
&\leq \text{trace}\left(\sum_{e \in \mathcal{E}} i_e^T R_e i_e\right),
\end{aligned}$$

where the first inequality is due to  $\eta(\mathcal{E}) \subset \bar{\mathcal{E}}$  and the summation involving positive numbers, the equality is a consequence of the definition of  $\bar{j}$ , and the last inequality follows from the fact that  $\bar{R}_{\eta(e)} \leq R_e$ ,  $\forall e \in \mathcal{E}$ . From this, Lemma 4.4.2, and the definition of effective resistance, we conclude that

$$\text{trace}(\mathbf{i}^T \bar{R}_{u,v}^{\text{eff}} \mathbf{i}) \leq \text{trace}(\mathbf{i}^T R_{u,v}^{\text{eff}} \mathbf{i}),$$

for every  $\mathbf{i} \in \mathbb{R}^{k \times k}$ , from which the result follows. ■

*Remark 4.6.1 (Role of edge directions).* Effective resistances are independent of the directions of the edges in the graph. Reversing the direction of an edge  $e$  simply reverses the sign of the current  $i_e$  on that edge. It follows from (4.6) that the effective resistance between any two nodes is unaffected by the edge-directions. Rayleigh's monotonicity offers further evidence of this fact. Therefore, from now on we use  $\mathcal{G} \subset \bar{\mathcal{G}}$  to denote that  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$ . Note that the graph partial order defined in Assumption 4.5.1 can now be understood to mean graph embedding; the results of Theorem 4.5.2 do not change in doing so.

It follows from the electrical analogy Theorem 4.5.1 that, although a measurement graph is directed because of the need to distinguish between a measurement of  $x_u - x_v$  and that of  $x_v - x_u$ , the BLUE covariances are independent of edge directions. □

### 4.6.2 Other properties

The next result shows that if the scalar effective resistances in an electrical network with 1-Ohm resistors is known, then the generalized effective resistances upon putting a single matrix resistance on every edge can be deduced from them. Its proof is provided in Section 4.9.

**Lemma 4.6.1.** *For a graph  $\mathcal{G}$  with finite maximum node degree, let  $r_{u,v}^{\text{eff}}$  denote the scalar effective resistance between two nodes  $u$  and  $v$  in a scalar electrical network  $(\mathcal{G}, 1)$  that has 1-Ohm resistors on every edge of the graph  $\mathcal{G}$ . Let  $(\mathcal{G}, R_o)$  be a generalized electrical network constructed from the same graph  $\mathcal{G}$  by assigning a generalized resistance  $R_o \in \mathbb{S}^{k+}$  to every edge of  $\mathcal{G}$ . Then,*

$$R_{u,v}^{\text{eff}} = r_{u,v}^{\text{eff}} R_o. \quad \square$$

It turns out that matrix-valued effective resistances obey the triangle inequality. It is known that scalar effective resistance obeys triangle inequality, and is therefore also referred to as the “resistance distance” [106]. Although the result in [106] was proved only for finite networks, it is not hard to extend it to infinite networks as well. Application of Lemma 4.6.1 then leads to the following simple extension of the triangle inequality to generalized networks with constant resistances on every edge.

**Lemma 4.6.2 (Triangle Inequality).** *Let  $(\mathcal{G}, R_o)$  be a generalized electrical network satisfying Assumption 4.2.1 with a constant resistance  $R_o \in \mathbb{S}^{k+}$  on every edge of  $\mathcal{G}$ . Then, for every triple of nodes  $u, v, w$  in the network,*

$$R_{u,w}^{\text{eff}} \leq R_{u,v}^{\text{eff}} + R_{v,w}^{\text{eff}}. \quad \square$$

The next result states that we can replace parallel edges by a single edge of appropriate resistance so that the effective resistances in the network are unchanged.

Recall that two edges  $e_1$  and  $e_2$  are said to be parallel if they are incident on the same set of nodes (irrespective of the direction of the edges). A similar result was stated without proof for finite graphs in Section 2.2.3. The proof of the proposition is provided in Section 4.9.

We assume that the edge set  $\mathcal{E}$  is specified as  $\mathcal{E} = \{\mathcal{E}_1, \mathcal{E}_2, \dots\}$ , such that if  $e \in \mathcal{E}_j$  for some  $j$ , then all the edges parallel to  $e$  also belong to  $\mathcal{E}_j$ .

**Proposition 4.6.1 (Parallel Resistors).** *Consider a generalized electrical network  $(\mathcal{G}, P)$  satisfying Assumption 4.2.1. Let  $\mathcal{E}_j \subset \mathcal{E}, j = 1, \dots$  be subsets of parallel edges as described above. The effective resistance between every pair of nodes remain the same if every set  $\mathcal{E}_j$  is replaced by a single edge  $e_j$  with edge resistance  $R_{e_j}$  that is specified by*

$$R_{e_j}^{-1} := \sum_{e \in \mathcal{E}_j} R_e^{-1}. \quad \square$$

The reader may notice that the above rule is simply the application of the parallel resistance formula to generalized resistances. The proof of the result is provided in Section 4.9.

### 4.6.3 Approximating infinite network currents

The next theorem shows that currents and effective resistances in an infinite network can be approximated with arbitrary accuracy by those in a sufficiently large but finite subnetwork. A similar result for the usual scalar electrical networks was established by Flanders [103, 104]. The proof of the theorem, which is inspired by [103], is provided in Section 4.9.

**Theorem 4.6.2 (Finite Approximation).** *Let  $(\mathcal{G}, R)$  be a network satisfying Assumption 4.2.1,  $\{\mathcal{G}_{(n)}\}$  a nested sequence of finite graphs satisfying Assump-*

tion 4.5.1, and  $u, v$  two arbitrary nodes that appear in every graph  $\mathcal{G}_{(n)}$ . For an arbitrary  $\mathbf{i} \in \mathbb{R}^{k \times k}$ , let  $i$  and  $i^{(n)}$  denote the currents from node  $u$  to node  $v$  in the infinite network  $(\mathcal{G}, R)$  and in the finite network  $(\mathcal{G}_{(n)}, R)$ , respectively, with intensity  $\mathbf{i}$ . Then,

$$\lim_{n \rightarrow \infty} i^{(n)} = i,$$

where convergence is in the  $\mathcal{H}_R$ -norm. In addition, denoting by  $R_{u,v}^{\text{eff}}$  and  $R_{u,v}^{\text{eff}(n)}$  the effective resistances between nodes  $u$  and  $v$  in the networks  $(\mathcal{G}, R)$  and  $(\mathcal{G}_{(n)}, R)$ , respectively, we have

$$\lim_{n \rightarrow \infty} R_{u,v}^{\text{eff}(n)} = R_{u,v}^{\text{eff}}. \quad \square$$

This result will be instrumental in showing that the BLU estimator error covariances in large finite networks converges to the effective resistance in the limiting infinite network.

#### 4.6.4 Electrical analogy for finite networks

In a *finite* measurement network, the BLUE covariance of a node variable  $x_u$  is the same as the generalized effective resistance between  $u$  and  $o$  in the corresponding electrical network, which is stated in the next theorem.

**Theorem 4.6.3 (Finite Electrical Analogy).** *Let  $(\mathcal{G}, P)$  be a measurement network with a finite weakly connected graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  and an edge-covariance function  $P : \mathcal{E} \rightarrow \mathbb{S}^{k+}$ , with node  $o$  as the reference node. For every node  $u \in \{\mathcal{V} \setminus o\}$ , the following statements hold.*

1. *The BLU estimator  $C$  of  $x_u$  in the finite measurement network  $(\mathcal{G}, P)$  is equal to the current  $i$  with identity intensity  $I_k$  in the generalized electrical*

network  $(\mathcal{G}, P)$  from  $u$  to  $o$ :

$$C = i.$$

2. The covariance  $\Sigma_{u,o}$  of the BLU estimation error  $x_u - \hat{x}_u$  is equal to the effective resistance  $R_{u,o}^{\text{eff}}$  between the node  $u$  and the reference node  $o$ :

$$\Sigma_{u,o} = R_{u,o}^{\text{eff}} \quad \square$$

It was shown in Section 2.2.1 that the BLU covariance of the vector of all node variables is given by the inverse of the Dirichlet Laplacian  $\mathcal{L}$  (see Theorem 2.2.1). It follows from the result above that the effective resistances in a finite network are the  $k \times k$  blocks on the diagonal of  $\mathcal{L}^{-1}$ . This is stated formally in the next corollary.

**Corollary 4.6.1.** *Consider a finite generalized electrical network  $(\mathcal{G}, R)$  satisfying Assumption 4.2.1 where  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  consists of  $n$  nodes, of which one is a reference node. Let the reference node be numbered as the  $n^{\text{th}}$  node and the others be numbered as  $1, \dots, n-1$ . Then, the effective resistance between the node with index  $u$  and the reference node  $n$  is given by*

$$R_{u,n}^{\text{eff}} = (\phi_u^T \otimes I_k) \mathcal{L}^{-1} (\phi_u \otimes I_k)$$

where  $\phi_u \in \mathbb{R}^{n-1}$  has all zeros except an 1 at the  $u^{\text{th}}$  location, and  $\mathcal{L}$  is the Dirichlet Laplacian of  $\mathcal{G}$  w.r.t. the boundary  $\{n\}$  and edge-weights  $R_e^{-1}$ .  $\square$

*Proof of Theorem 4.6.3.* From Proposition 2.2.1 on the characterization of optimal estimators of node variables in finite graphs and the definition of energy dissipation (4.5), we see that in a finite network  $(\mathcal{G}, P)$  with reference node  $o$ , the

BLU estimator  $C$  of node variable  $x_u$  is given by

$$C = \arg \min \|j\|$$

subject to:  $j$  is a flow of intensity  $I_k$  from  $u$  to  $o$ .

Comparing with the electrical network problem, we conclude from Theorem 4.4.1 that the BLU estimator  $C$  of  $x_u$  is the current  $i$  of intensity  $I_k$  from  $u$  to  $o$  in the generalized electrical network  $(\mathcal{G}, P)$ , which proves the first statement.

Since  $C = i$ , it follows from Unbiased Estimator Lemma 2.2.2 that the covariance of  $x_u$ 's BLU estimation error is given by

$$\Sigma_{u,o} = \sum_{e \in \mathcal{E}} i^T P_e i_e = R_{u,o}^{\text{eff}},$$

where the second inequality follows from (4.6), which proves the second statement. ■

## 4.7 Proof of the main results

First we prove the BLUE convergence Theorem 4.5.2, using the tools developed so far.

*Proof of Theorem 4.5.2.* We will prove the statements of the theorem in reverse order.

Since the sequence of BLU covariances  $\Sigma_{u,o}^{(n)}$  is the same as the sequence of effective resistances  $R_{u,o}^{\text{eff}(n)}$  (Finite Electrical Analogy Theorem 4.6.3), and the sequence  $R_{u,o}^{\text{eff}(n)}$  converges to the effective resistance  $R_{u,o}^{\text{eff}}$  in the infinite network (Finite Approximation Theorem 4.6.2), we have

$$\Sigma_{u,o}^{(n)} = R_{u,o}^{\text{eff}(n)} \rightarrow R_{u,o}^{\text{eff}}.$$

Moreover, by the construction of the nested sequence  $\{\mathcal{G}^{(n)}\}$ , if  $n_1 \leq n_2$ , then  $\mathcal{G}^{(n_1)} \subset \mathcal{G}^{(n_2)}$ , and so by Rayleigh's monotonicity law (Theorem 4.6.1),

$$\Sigma_{u,o}^{(1)} \geq \Sigma_{u,o}^{(2)} \geq \dots,$$

from which the third statement of the theorem follows.

The BLU estimator  $C^{(n)}$  of  $x_u$  in the finite network  $(\mathcal{G}_{(n)}, P)$  is equal to the current  $i^{(n)}$  in the generalized electrical network  $(\mathcal{G}_{(n)}, P)$  (Finite Electrical Analogy Theorem 4.6.3), and the currents  $i^{(n)}$  converge to the unique current  $i$  in the electrical network  $(\mathcal{G}, R)$  (Finite Approximation Theorem 4.6.2). Therefore

$$C^{(n)} = i^{(n)} \rightarrow i =: C,$$

where the convergence is in the  $\mathcal{H}_P$ -norm. This proves the second statement.

By definition of the BLU estimator, we get

$$\begin{aligned} \hat{x}_u^{(n)} &= \sum_{(p,q) \in \mathcal{E}^{(n)}} C_{p,q}^{(n)T} (x_p - x_q + \epsilon_{p,q}) \\ &= x_u + \sum_{(p,q) \in \mathcal{E}^{(n)}} C_{p,q}^{(n)T} \epsilon_{p,q}, \end{aligned} \tag{4.9}$$

where the second equality follows from unbiasedness, since otherwise the expectation of the left hand side would not be equal to  $x_u$ . Let  $n < l$ , so that from Nested Sequence Assumption 4.5.1,  $\mathcal{G}_{(n)} \subset \mathcal{G}_{(l)}$ . It follows from the uncorrelated-ness of the  $\epsilon$ 's and (4.9) that

$$\mathbb{E}[(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})^T] = \sum_{e \in \mathcal{E}^{(l)}} (C_e^{(l)} - C_e^{(n)})^T P_e (C_e^{(l)} - C_e^{(n)}),$$

where we have used the convention that  $C_e^{(n)} = 0$  if  $e \in \mathcal{E}^{(l)} \setminus \mathcal{E}^{(n)}$ . This leads to

$$\text{trace} \left( \mathbb{E}[(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})^T] \right) = \|C^{(l)} - C^{(n)}\|^2,$$

where  $\|\cdot\|$  is the  $\mathcal{H}_P$ -norm. Since  $C^{(n)} \rightarrow i$ ,  $\|C^{(l)} - C^{(n)}\| \rightarrow 0$  as  $n, m \rightarrow \infty$ . Therefore,

$$\lim_{n \rightarrow \infty} \sup_{l \geq n} \text{trace} \left( \mathbb{E}[(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})(\hat{x}_u^{(l)} - \hat{x}_u^{(n)})^T] \right) = 0. \quad (4.10)$$

We recall that a sequence of random variables  $\{\eta_n\}$  converges in the mean square sense if and only if (proposition 6.3 in [107])

$$\lim_{n, l \rightarrow \infty} \sup_{l \geq n} \mathbb{E}[|\eta_l - \eta_n|^2] = 0.$$

Therefore, the sequence of random vectors  $\hat{x}_u^{(n)}$  converge entry-wise in the mean square sense. This proves the first statement and completes the proof.  $\blacksquare$

It was shown in the previous section that for finite measurement networks, the effective resistance is the same as the BLUE covariance. The Electrical Analogy Theorem 4.5.1 is an extension of this analogy to infinite networks, which follows as a consequence of Theorem 4.5.2.

*Proof of Theorem 4.5.1.* When the graph  $\mathcal{G}$  is finite, the statement of the theorem follows from Finite Electrical Analogy Theorem 4.6.3.

When  $\mathcal{G}$  is infinite, consider a sequence  $\{\mathcal{G}_{(n)}\}$  of nested finite subgraphs of the infinite graph  $\mathcal{G}$  that satisfies the Nested Sequence Assumption 4.5.1. We know from the BLUE convergence Theorem 4.5.2 that the sequence of BLU estimation error co-variance matrices  $\Sigma_{u,o}^{(n)}$  converges monotonically to the effective resistance  $R_{u,o}^{\text{eff}}$ , i.e.,

$$\Sigma_{u,o}^{(1)} \geq \Sigma_{u,o}^{(2)} \geq \dots, \text{ and } \lim_{n \rightarrow \infty} \Sigma_{u,o}^{(n)} = R_{u,o}^{\text{eff}}.$$

It is straightforward to show that  $R_{u,o}^{\text{eff}} = \inf\{\Sigma_{u,o}^{(n)}, n \in \mathbb{N}\}$  according to the matrix infimum definition (4.2). That is,  $R_{u,o}^{\text{eff}} \leq \Sigma_{u,o}^{(n)} \forall n \in \mathbb{N}$ , and for every  $\epsilon > 0$ ,  $\exists n$

such that  $R_{u,o}^{\text{eff}} + \epsilon I_k > \Sigma_{u,o}^{(n)}$ . Now we will show that  $R_{u,o}^{\text{eff}}$  is also the infimum of the set

$$S := \{\Sigma_{u,o}(\mathcal{G}_{\text{finite}}), \mathcal{G}_{\text{finite}} \subset \mathcal{G}, \mathcal{G}_{\text{finite}} \text{ is finite}\}.$$

If  $\mathcal{G}_{\text{finite}}$  is an arbitrary finite subgraph of the infinite graph  $\mathcal{G}$ , then  $\exists n \in \mathbb{N}$  such that  $\mathcal{G}_{\text{finite}} \subset \mathcal{G}_{(n)}$ . By the Finite Electrical Analogy Theorem 4.6.3 and Rayleigh's monotonicity law Theorem 4.6.1, we have  $\Sigma_{u,o}^{(n)} \leq \Sigma_{u,o}(\mathcal{G}_{\text{finite}})$ . Since  $R_{u,o}^{\text{eff}} \leq \Sigma_{u,o}^{(n)}$ , we have that  $R_{u,o}^{\text{eff}}$  is a lower bound of the set  $S$  defined above.

To show that  $R_{u,o}^{\text{eff}}$  is the largest lower bound, pick an  $\epsilon > 0$  and pick  $m \in \mathbb{N}$  such that  $R_{u,o}^{\text{eff}} + \epsilon I_k > \Sigma_{u,o}^{(m)}$ . Such an  $m$  exists since  $R_{u,o}^{\text{eff}}$  is the infimum of  $\{\Sigma_{u,o}^{(n)}\}$ . Now pick any finite subgraph  $\mathcal{G}_{\text{finite}}$  of  $\mathcal{G}$  such that  $\mathcal{G}_{\text{finite}} \supset \mathcal{G}^{(m)}$ . From the electrical analogy for finite networks and Rayleigh's Monotonicity Law, we have  $\Sigma_{u,o}^{(m)} \geq \Sigma_{u,o}(\mathcal{G}_{\text{finite}})$ . Thus, for every  $\epsilon > 0$ , we can find a finite subgraph  $\mathcal{G}_{\text{finite}}$  of  $\mathcal{G}$  such that  $R_{u,o}^{\text{eff}} + \epsilon I_k \geq \Sigma_{u,o}(\mathcal{G}_{\text{finite}})$ . We therefore have  $R_{u,o}^{\text{eff}} = \inf S$ .

This proves that the infimum  $\Sigma_{u,o}$  of (4.1) is well defined, and is equal to the effective resistance  $R_{u,o}^{\text{eff}}$ , which concludes the proof. ■

## 4.8 Comments and open problems

In this chapter we provided a formal justification for using infinite graphs as valid approximation of large but finite graphs, and established conditions under which this approximation is valid. For example, the condition of bounded degree of the graph (see Assumption 4.2.1) is seen to be important in the proof of convergence results.

On the other hand, certain important classes of graphs, such as scale free graphs [108] and random geometric graphs [109], have unbounded degree. In a

random geometric graph of  $n$  nodes, the average degree of a node has to be  $\Omega(\log n)$  to ensure connectivity with high probability [109]. Scale-free graphs exhibit a heavy-tailed degree distribution so that some nodes have very large degrees with very small probability. As a result, it is not possible – at least within the confines of the techniques used here – to study these graphs by examining the behavior of the limiting infinite graph. It is unclear what will it take to study effective resistance in infinite graphs with unbounded degree.

The BLUE convergence Theorem 4.5.2 points to an interesting direction for designing distributed algorithms. The theorem shows that when an infinite number of measurements are available, the estimate of a node variable based on a finite subset of the measurements can be arbitrarily close to the estimate that can be obtained by using all the available measurements. This suggests that using small subsets of available measurements can give us estimates that are quite close to the optimal. How to design distributed algorithms to take advantage of this feature is an open question.

## 4.9 Technical proofs

We first introduce some terminology. Define a norm for all node-functions  $\omega : \mathcal{V} \rightarrow \mathbb{R}^{k \times k}$  as

$$\|\omega\| = \left( \sum_{u \in \mathcal{V}} \text{trace}(\omega_u^T \omega_u) \right)^{\frac{1}{2}} = \left( \sum_{u \in \mathcal{V}} \|\omega_u\|_F^2 \right)^{\frac{1}{2}}, \quad (4.11)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm of a matrix, and a linear vector space  $S_{\mathcal{V}}$  as the space of all bounded node-functions with respect to the above defined

norm:

$$S_{\mathcal{V}} = \{\omega : \mathcal{V} \rightarrow \mathbb{R}^{k \times k} \mid \|\omega\| < \infty\}. \quad (4.12)$$

For an infinite network  $(\mathcal{G}, R)$ , we introduce the *incidence operator*  $\mathcal{A} : \mathcal{H}_R \rightarrow S_{\mathcal{V}}$ , which is defined by the transformation:

$$(\mathcal{A}j)_u = \sum_{e \in \mathcal{E}} a_{u,e} j_e, \quad j \in \mathcal{H}_R, \quad (4.13)$$

where  $a_{u,e}$  is nonzero if and only if the edge  $e$  is incident on the node  $u$  and when nonzero,  $a_{u,e} = -1$  if the edge  $e$  is directed towards  $u$  and  $a_{u,e} = 1$  otherwise. The incidence operator  $\mathcal{A}$  is simply an extension to infinite graphs of the generalized incidence matrix defined in Section 2.2.1 [see (2.3)] for finite graphs. The series in (4.13) is absolutely convergent since it involves only a finite number of terms due to the bounded degree of  $\mathcal{G}$ .

We call a node-function  $\omega \in S_{\mathcal{V}}$  a *divergence* for the graph  $\mathcal{G}$  if  $\omega$  has finite support and  $\sum_{u \in \mathcal{V}} \omega_u = 0$ . One can view a divergence as an assignment of flow sources at a finite number of nodes of the graph so that total flow into the graph is equal to the total flow out of it.

An edge-function  $j \in \mathcal{H}_R$  is called a *flow in  $\mathcal{G}$  with divergence  $\omega \in S_{\mathcal{V}}$*  if  $\omega$  is a divergence in  $\mathcal{G}$  and  $j$  satisfies

$$\sum_{\substack{(u,v) \in \mathcal{E} \\ u = \bar{u}}} j_{u,v} - \sum_{\substack{(v,u) \in \mathcal{E} \\ u = \bar{u}}} j_{v,u} = \omega_{\bar{u}}, \quad \forall \bar{u} \in \mathcal{V}. \quad (4.14)$$

The condition (4.14) can be compactly represented as

$$\mathcal{A}j = \omega. \quad (4.15)$$

An edge-function  $j \in \mathcal{H}_R$  is called a *circulation in  $(\mathcal{G}, R)$*  if

$$\mathcal{A}j = 0. \quad (4.16)$$

In other words, a circulation is an element of  $\mathcal{H}_R$  that belongs to  $\mathcal{N}(\mathcal{A})$ , the null space of  $\mathcal{A}$ .

First we show that the linear operator  $\mathcal{A} : \mathcal{H}_R \rightarrow S_{\mathcal{V}}$  defined above is bounded. Since for each  $u \in \mathcal{V}$ ,  $(\mathcal{A}j)_u \in \mathbb{R}^{k \times k}$ , we have

$$\|(\mathcal{A}j)_u\|_F^2 = \left\| \sum_{e \in \mathcal{E}_u} a_{ue} j_e \right\|_F^2 \leq \sum_{e \in \mathcal{E}_u} \|j_e\|_F^2$$

where  $\mathcal{E}_u$  is the set edges in  $\mathcal{E}$  that are incident on  $u$ . It can be shown from the relationship between the Frobenius norm and the singular values of a matrix that for every edge  $e \in \mathcal{E}$ , we have  $\|j_e\|_F^2 \leq \frac{1}{\lambda_{\min}} \text{trace}(j_e^T R_e j_e)$ , where  $\lambda_{\min}$  is the uniform lower bound on the smallest eigenvalue of  $R_e$ ,  $\forall e \in \mathcal{E}$ . Existence of a positive  $\lambda_{\min}$  is guaranteed by Assumption 4.2.1. Since the above is true for every  $u \in \mathcal{V}$ , from (4.11) we get

$$\begin{aligned} \|\mathcal{A}j\|^2 &= \sum_{u \in \mathcal{V}} \|(\mathcal{A}j)_u\|_F^2 \leq \frac{1}{\lambda_{\min}} \sum_{u \in \mathcal{V}} \sum_{e \in \mathcal{E}_u} \text{trace}(j_e^T R_e j_e) \\ &\leq \frac{d_{\max}}{\lambda_{\min}} \sum_{e \in \mathcal{E}} \text{trace}(j_e^T R_e j_e) = \frac{d_{\max}}{\lambda_{\min}} \|j\|^2, \end{aligned}$$

where  $d_{\max}$  is the largest degree of the nodes of the graph  $\mathcal{G}$ , which is finite by Assumption 4.2.1. It follows that

$$\|\mathcal{A}\| \leq \sqrt{\frac{d_{\max}}{\lambda_{\min}}},$$

which shows that  $\mathcal{A}$  is bounded.

Now we are ready to prove the Infinite Current Theorem 4.4.1.

*Proof of Theorem 4.4.1.* We first prove that among the flows in  $\mathcal{H}_R$  that are limits of finite support flows, the flow with the minimum dissipated energy exists and is unique, and that this flow is a current. Then we show that there can be only one such current.

For a flow of intensity  $\mathbf{j}$  that is injected at  $u$  and extracted at  $v$ , the corresponding divergence  $\bar{\omega}$  is given by  $\bar{\omega}_u = \mathbf{j}$ ,  $\bar{\omega}_v = -\mathbf{j}$  and  $\bar{\omega}_p = 0$  for all  $p \in \mathcal{V} \setminus \{u, v\}$ . Pick a path  $\mathcal{P}$  from  $u$  to  $v$ , and construct a flow  $j^{\text{path}}$  of intensity  $\mathbf{j}$  from  $u$  to  $v$  along  $\mathcal{P}$  as follows:

$$j_e^{\text{path}} = \begin{cases} \mathbf{j} & e \in \mathcal{P}, \vec{e} = \vec{\mathcal{P}} \\ -\mathbf{j} & e \in \mathcal{P}, \vec{e} \neq \vec{\mathcal{P}} \\ 0 & e \notin \mathcal{P} \end{cases}$$

It is easy to see that  $j$  is a finite support edge-function in  $\mathcal{H}_R$  that satisfies the constraint equation  $\mathcal{A}j = \bar{\omega}$ . All flows satisfying this constraint lie in the linear variety  $j^{\text{path}} + \mathcal{N}(\mathcal{A})$ , where  $\mathcal{N}(\mathcal{A})$  is the null space of  $\mathcal{A}$ . Since  $\mathcal{A}$  is a bounded linear operator, its null space is closed. As a result,  $\mathcal{N}(\mathcal{A})$ , which is the space of all circulations, is a Hilbert space. Consider the subspace of  $\mathcal{N}(\mathcal{A})$  that consists of all finite support circulations, and denote it by  $\mathcal{N}_F(\mathcal{A})$  (“ $F$ ” for finite support). Its closure  $\overline{\mathcal{N}_F(\mathcal{A})}$  is a closed subspace of the Hilbert space  $\mathcal{N}(\mathcal{A})$ . By the Projection Theorem applied to linear varieties (Theorem 1 in section 3.10 of [62]), there exists a unique edge-function in  $j^{\text{path}} + \overline{\mathcal{N}_F(\mathcal{A})}$  of minimum norm, which we call  $i$ , and which is orthogonal to  $\overline{\mathcal{N}_F(\mathcal{A})}$ .

Since  $i - j^{\text{path}} \in \overline{\mathcal{N}_F(\mathcal{A})}$ , there exists a sequence of finite support circulations  $c^{(n)}$  such that  $c^{(n)} \rightarrow (i - j^{\text{path}})$ , where the convergence is in  $\mathcal{H}_R$  norm. Define  $j^{(n)} := j^{\text{path}} + c^{(n)}$ , so that by construction, each  $j^{(n)}$  is a finite support flow of intensity  $\mathbf{i}$  from  $u$  to  $v$ , and  $j^{(n)} \rightarrow i$  in  $\mathcal{H}_R$ . This establishes the existence and uniqueness of the flow with minimum power dissipation that is the limit of a sequence of finite support flows.

Since  $i$  is orthogonal to  $\overline{\mathcal{N}_F(\mathcal{A})}$ ,

$$\langle i, c \rangle = 0 \quad (4.17)$$

for every every  $c \in \overline{\mathcal{N}_F(\mathcal{A})}$ . Declare the generalized potential drop across an edge  $e$  as  $R_e i_e$  to satisfy Ohm's law. If the graph has no loops, Kirchoff's loop law is trivially satisfied by these generalized potential drops. If the graph has loops, pick a loop  $\mathcal{C}$  and define a scalar edge-function  $f : \mathcal{E} \rightarrow \mathbb{R}$  as

$$f_e = \begin{cases} 1 & \text{if } e \in \mathcal{C} \text{ and } \vec{e} = \vec{\mathcal{C}}, \\ -1 & \text{if } e \in \mathcal{C} \text{ and } \vec{e} \neq \vec{\mathcal{C}}, \\ 0 & \text{if } e \notin \mathcal{C}. \end{cases}$$

Now define a finite support circulation  $c^*$  as  $c_e^* = f_e J$ , where  $J$  is an arbitrary  $k \times k$  matrix. We have

$$\begin{aligned} 0 &= \langle i, c^* \rangle = \sum_{e \in \mathcal{C}} \text{trace}(i_e^T R_e c_e^*) \\ &= \sum_{e \in \mathcal{C}} f_e \text{trace}(i_e^T R_e J) = \sum_{e \in \mathcal{C}} f_e \text{trace}(J^T R_e i_e) \\ &= \text{trace} \left[ J^T \left( \sum_{e \in \mathcal{C}} f_e R_e i_e \right) \right] \end{aligned}$$

Since this is true for arbitrary  $J$ , we must have

$$\sum_{e \in \mathcal{C}} [f_e (R_e i_e)] = 0, \quad (4.18)$$

which in turn must be true for every loop  $\mathcal{C}$ , since the arguments above can be repeated for every loop. Eq. (4.18) therefore shows that the *net potential drop* along every loop is 0. In other words, the generalized potential drops determined by  $i$  in accordance with Ohm's law satisfies Kirchoff's loop law. Construction of a generalized node potential function  $V$  is now trivial. Therefore  $i$  is a generalized current.

To prove uniqueness of the current, let  $i$  and  $\bar{i}$  be two currents from  $u$  to  $v$  with intensity  $\mathbf{i}$ . Define an edge-function  $d : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  as  $d_e := i_e - \bar{i}_e$ . We see that  $d \in \overline{\mathcal{N}_F(\mathcal{A})}$ . From linearity of the inner product,

$$\langle d, d \rangle = \langle i - \bar{i}, i - \bar{i} \rangle = \langle i, d \rangle - \langle \bar{i}, d \rangle = 0 - 0,$$

where the last equalities follows from (4.17), since by construction, both  $i$  and  $\bar{i}$  are currents. It follows that

$$\sum_{e \in \mathcal{E}} \text{trace}(d_e^T R_e d_e) = 0 \quad \Rightarrow \quad d_e = 0 \quad \forall e \in \mathcal{E},$$

since  $R_e > 0$  for all edges  $e \in \mathcal{E}$ . We therefore conclude that  $i = \bar{i}$ , which proves that the current  $i$  is unique.

To examine the uniqueness of potentials, suppose that  $V$  and  $\bar{V}$  are two potentials associated with the same current. Because of Ohm's Law, we conclude that

$$V_u - V_v = \bar{V}_u - \bar{V}_v \Rightarrow D_u = D_v, \forall (u, v) \in \mathcal{E},$$

where  $D = V - \bar{V}$ . Since  $\mathcal{G}$  is connected,  $D$  must be a constant, but is otherwise arbitrary. This shows that the node potentials are unique up to an additive constant.

If  $i$  is a current with intensity  $\mathbf{i}$  and  $\bar{i}$  is a current with intensity  $\bar{\mathbf{i}}$ , both from  $u$  to  $v$ , it can be shown in a straightforward manner that  $\alpha i + \beta \bar{i}$  is also a current with intensity  $\alpha \mathbf{i} + \beta \bar{\mathbf{i}}$  from  $u$  to  $v$ , from which the linearity from  $\mathbf{i}$  to  $i$  follows. A similar linearity proof also holds for the potential differences. ■

The corollary presented next is essentially a repetition of (4.17), but is restated because of its usefulness in several subsequent proofs.

**Corollary 4.9.1.** *A flow  $i$  is the generalized current in the network  $(\mathcal{G}, R)$  if and only if*

$$\langle i, c \rangle = 0$$

*for every circulation  $c \in \overline{\mathcal{N}_F(\mathcal{A})}$ .* □

Next we prove that the linear mapping between intensity and voltage drop between the source node and sink node is given by a  $k \times k$  matrix.

*Proof of Lemma 4.4.1.* For the current with intensity  $\mathbf{i}$  flowing from  $u$  to  $v$ , we define a divergence  $\omega$  as

$$\omega_p = 0 \quad \forall p \in \mathcal{V} \setminus \{u, v\}, \quad \omega_u = \mathbf{i}, \omega_v = -\mathbf{i}.$$

The flow constraint now becomes  $\mathcal{A}j = \omega$ . The current  $i$  is the flow that satisfies this constraint and minimizes the energy dissipation  $\sum_{e \in \mathcal{E}} \text{trace}(j_e^T R_e j_e)$ , as shown in Theorem 4.4.1. For every node  $p \in \mathcal{V}$ , the flow constraint becomes

$$(\mathcal{A}j)_p = \omega_p \Rightarrow \sum_{e \in \mathcal{E}_p} a_{p,e} j_e = \omega_p. \quad (4.19)$$

Recognizing that this is a  $k \times k$  matrix equation, we express it as  $k$  separate vector equations:

$$\sum_{e \in \mathcal{E}_p} a_{p,e} j_{e,l} = \omega_{p,l}, \quad l = 1, \dots, k,$$

where the second subscript  $l$  represents the  $l^{\text{th}}$  column of the corresponding matrix. It is easy to see that, for every  $l$ , the constraints on the  $l^{\text{th}}$  column of  $j_e$ 's depend only on the  $l^{\text{th}}$  column of  $\omega_p$ , and therefore on the  $l^{\text{th}}$  column of  $\mathbf{i}$ . As a result, the solution to this optimization problem is equivalent to solving  $k$  separate problems “minimize  $\sum_{e \in \mathcal{E}} j_{e,l}^T R_e j_{e,l}$  subject to  $\mathcal{A}j_l = \omega_l$ ”, for  $l = 1, \dots, k$ , where the edge

function  $j_l$  and the node function  $\omega_l$  are now vector-valued:  $j_l : \mathcal{E} \rightarrow \mathbb{R}^k$ ,  $\omega_l : \mathcal{V} \rightarrow \mathbb{R}^k$ , the spaces  $\mathcal{H}_R$  and  $S_{\mathcal{V}}$  are appropriately redefined, and the incidence operator  $\mathcal{A}$  has the same definition as in (4.13) with respect to the new spaces  $\mathcal{H}_R, S_{\mathcal{V}}$ . Because of column-wise independence of the current on the intensities, the matrix current on every edges is obtained by stacking the  $k$  vector-valued currents on that edge as columns. For every vector-valued current intensity  $\mathbf{i}_l, l = 1, \dots, k$ , we obtain a corresponding vector-valued potential difference  $V_{u,l} - V_{v,l}$ . Again, the matrix-valued potential difference  $V_u - V_v$  resulting from the original problem consists of the  $k$  columns that are the vector-valued potential difference  $V_{u,l} - V_{v,l}$  resulting from the  $k$  separate optimization problems described above.

These  $k$  separate optimization problems can be solved to determine the vector-valued edge currents in the same manner that the single optimization problem was solved in the proof of Theorem 4.4.1 to determine the matrix valued edge currents. In fact, only one of these  $k$  problems needs to be solved. To understand why, we first note that the linearity between the matrix valued quantities  $\mathbf{i}$  and  $V_u - V_v$  that was established in Theorem 4.4.1 will be retained between the corresponding vector-valued quantities. Specifically, when a vector-valued current  $i_l$  flows from  $u$  to  $v$  with vector intensity  $\mathbf{i}_l$ , the vector-valued voltage drop  $V_{u,l} - V_{v,l}$  will be a linear function of the vector intensity  $\mathbf{i}_l$ , which will be in general a  $k \times k$  matrix. Let  $R_{u,v}^{\text{eff}} \in \mathbb{R}^{k \times k}$  be this matrix. Then,

$$V_{u,l} - V_{v,l} = R_{u,v}^{\text{eff}} \mathbf{i}_l, \quad \forall \mathbf{i}_l \in \mathbb{R}^k. \quad (4.20)$$

From linearity, the same is true for every  $l = 1, \dots, k$ . Stacking together the  $k$  columns in (4.20), for  $l = 1, \dots, k$ , we get  $V_u - V_v = R_{u,v}^{\text{eff}} \mathbf{i}$ , which proves that the linear mapping between matrix intensity  $\mathbf{i}$  and matrix-valued potential drop  $V_u - V_v$  is the  $k \times k$  matrix  $R_{u,v}^{\text{eff}}$ . ■

*Proof of Lemma 4.4.2.* Pick a path  $\mathcal{P}$  from  $u$  to  $v$ , and construct a flow  $j^{\text{path}}$  of intensity  $\mathbf{j}$  from  $u$  to  $v$  along  $\mathcal{P}$  as follows:

$$j_e^{\text{path}} = \begin{cases} \mathbf{j} & e \in \mathcal{P}, \vec{e} = \vec{\mathcal{P}} \\ -\mathbf{j} & e \in \mathcal{P}, \vec{e} \neq \vec{\mathcal{P}} \\ 0 & e \notin \mathcal{P} \end{cases}$$

The assumed properties of  $j$  imply that  $j \in j^{\text{path}} + \overline{\mathcal{N}_F(A)}$ . Let  $j^{(n)}$  be a sequence of finite support flows in  $(\mathcal{G}, R)$  that converge to the flow  $j$ , i.e.,  $j^{(n)} \rightarrow j$  in  $\mathcal{H}_R$ . Define

$$\begin{aligned} c &:= j - j^{\text{path}} \\ c^{(n)} &:= j^{(n)} - j^{\text{path}}. \end{aligned}$$

The function  $c \in \mathcal{H}_R$  is a circulation since it is the difference between two flows of the same intensity between the same two nodes. Moreover,  $\{c^{(n)}\}$  is a sequence of finite-support circulations that converge to  $c$  in  $\mathcal{H}_R$ . Now, since  $c^{(n)}$  is a finite support circulation, from Corollary 4.9.1,  $\langle i, c^{(n)} \rangle = \langle i, j^{\text{path}} - j^{(n)} \rangle = 0$  for every  $n$ , and therefore

$$\lim_{n \rightarrow \infty} \langle i, j^{\text{path}} - j^{(n)} \rangle = 0,$$

Using linearity and continuity of the inner product, we therefore conclude that

$$\begin{aligned} \lim_{n \rightarrow \infty} \langle i, j^{(n)} \rangle &= \langle i, j^{\text{path}} \rangle \Rightarrow \langle i, j \rangle = \langle i, j^{\text{path}} \rangle \\ \Rightarrow \sum_{e \in \mathcal{E}} \text{trace}(i_e^T R_e j_e) &= \sum_{e \in \mathcal{P}} \text{trace}((R_e i_e)^T j_e^{\text{path}}) = \text{trace}((V_u - V_v)^T \mathbf{j}) \end{aligned} \quad (4.21)$$

Since  $i, j \in \mathcal{H}_R$ , denoting the  $s^{\text{th}}$  column of  $i_e$  by  $i_{s,e}$  and the  $t^{\text{th}}$  column of  $j_e$  by  $j_{t,e}$ , we can show from (4.21) using straightforward algebraic manipulation that

$$q_{s,t} := \sum_{e=1}^{\infty} i_{s,e}^T R_e j_{t,e} < \infty. \quad \forall s, t = \{1, \dots, k\},$$

and that the series converges absolutely for every  $s$  and  $t$ . Define the matrix  $Q$  by  $[Q]_{s,t} = q_{s,t}$ . Since the series converges, for every  $\epsilon > 0$ , we can choose  $N$  large enough such that

$$\left\| \sum_{e=1}^N i_e^T R_e j_e - Q \right\| < \epsilon,$$

where  $\|\cdot\|$  represents any matrix norm. We thus conclude that since  $i, j \in \mathcal{H}_R$ , the series  $\sum_{e \in \mathcal{E}} i_e^T R_e j_e$  converges absolutely to a  $k \times k$  matrix. Since (4.21) holds for an arbitrary  $\mathbf{j}$ , it can be shown in a straightforward manner that the series  $\sum_{e \in \mathcal{E}} i_e^T R_e j_e$  must converge to  $(V_u - V_v)^T \mathbf{j}$ . Therefore we get the desired result

$$\sum_{e \in \mathcal{E}} i_e^T R_e j_e = (V_u - V_v)^T \mathbf{j}. \quad \blacksquare$$

We now prove the Finite Approximation Theorem 4.6.2.

*Proof of Theorem 4.6.2.* For every  $\epsilon > 0$ , we can find a finite-support flow  $j^{(n)}$  from  $u$  to  $v$  of intensity  $\mathbf{i}$  such that

$$\|i - j^{(n)}\| < \epsilon, \quad (4.22)$$

which follows from the characterization of the current  $i$  in Theorem 4.4.1. Pick a finite subgraph  $\mathcal{G}^{(n)} = (\mathcal{V}^{(n)}, \mathcal{E}^{(n)})$  of  $\mathcal{G}$  from the nested sequence  $\{\mathcal{G}^{(n)}\}$  such that the support of  $j^{(n)}$  lies in  $\mathcal{G}^{(n)}$  (i.e., the edges on which  $j^{(n)}$  is not zero are in  $\mathcal{E}^{(n)}$ ). Note that by construction  $u, v \in \mathcal{V}^{(n)}$ . Denoting by  $i^{(n)}$  the current in  $(\mathcal{G}^{(n)}, R)$ , it follows from Corollary 4.9.1 that for a circulation  $c^{(n)}$  whose support lies in  $\mathcal{G}^{(n)}$ ,

$$\begin{aligned} \langle i^{(n)}, c^{(n)} \rangle &= 0, \text{ and } \langle i, c^{(n)} \rangle = 0. \\ \Rightarrow |\langle i - j^{(n)}, c^{(n)} \rangle| &= |\langle i, c^{(n)} \rangle - \langle j^{(n)}, c^{(n)} \rangle| \\ &= |\langle i^{(n)}, c^{(n)} \rangle - \langle j^{(n)}, c^{(n)} \rangle| \\ &= |\langle i^{(n)} - j^{(n)}, c^{(n)} \rangle|. \end{aligned}$$

Pick  $c^{(n)} = i^{(n)} - j^{(n)}$ , which, being a difference of two finite support flows from  $u$  to  $v$  with the same intensity, is a finite support circulation. Furthermore, its support lies in  $\mathcal{G}^{(n)}$  since both  $i^{(n)}$  and  $j^{(n)}$  have their support in  $\mathcal{G}^{(n)}$ . For this choice of  $c^{(n)}$  in the equation above, we get

$$\begin{aligned} |(i - j^{(n)}, i^{(n)} - j^{(n)})| &= \|i^{(n)} - j^{(n)}\|^2 \\ \Rightarrow \|i^{(n)} - j^{(n)}\|^2 &\leq \|i - j^{(n)}\| \|i^{(n)} - j^{(n)}\|, \end{aligned}$$

from the Cauchy Schwarz inequality. Therefore,

$$\|i^{(n)} - j^{(n)}\| \leq \|i - j^{(n)}\| < \epsilon,$$

from (4.22). From the triangle inequality, we now get

$$\|i - i^{(n)}\| \leq \|i - j^{(n)}\| + \|i^{(n)} - j^{(n)}\| < 2\epsilon,$$

which proves the statement that  $i^{(n)} \rightarrow i$  in  $\mathcal{H}_R$ .

To prove the convergence of the effective resistances, pick an arbitrary  $\mathbf{i} \in \mathbb{R}^{k \times k}$  and let  $i$  and  $i^{(n)}$  be the currents with intensity  $\mathbf{i}$  from  $u$  to  $v$  in  $(\mathcal{G}, R)$  and  $(\mathcal{G}^{(n)}, R)$ , respectively. It follows from Lemma 4.4.2 that

$$\begin{aligned} \sum_{e \in \mathcal{E}} i_e^T R_e i_e &= \mathbf{i}^T R_{u,v}^{\text{eff}} \mathbf{i}, \\ \sum_{e \in \mathcal{E}} i_e^{(n)T} R_e i_e^{(n)} &= \sum_{e \in \mathcal{E}^{(n)}} i_e^{(n)T} R_e i_e^{(n)} = \mathbf{i}^T R_{u,v}^{\text{eff}(n)} \mathbf{i}, \\ \sum_{e \in \mathcal{E}} i_e^T R_e i_e^{(n)} &= \mathbf{i}^T R_{u,v}^{\text{eff}} \mathbf{i}, \end{aligned}$$

where the last equality uses the fact that  $i^{(n)}$  is a flow in  $\mathcal{G}$  with intensity  $\mathbf{i}$  (though not a current). Therefore,

$$\sum_{e \in \mathcal{E}} \text{trace}((i - i_e^{(n)})^T R_e (i_e - i_e^{(n)})) = \text{trace}(\mathbf{i}^T (R_{u,v}^{\text{eff}(n)} - R_{u,v}^{\text{eff}}) \mathbf{i})$$

Since  $i \rightarrow i^{(n)}$  in  $\mathcal{H}_R$ , the left hand side goes to 0 as  $n \rightarrow \infty$ . Since this is true for arbitrary  $\mathbf{i}$ ,  $R_{u,v}^{\text{eff}(n)} \rightarrow R_{u,v}^{\text{eff}}$ . ■

*Proof of Lemma 4.6.1.* Let  $i^s : \mathcal{E} \rightarrow \mathbb{R}$  be the scalar current in  $(\mathcal{G}, 1)$  of unit scalar intensity from  $u$  to  $v$ . It follows from (4.6) that

$$r_{u,v}^{\text{eff}} = \sum_{e \in \mathcal{E}} (i_e^s)^2.$$

We first claim that the matrix current  $i$  in  $(\mathcal{G}, R_o)$  of intensity  $I_k$  from  $u$  to  $v$  is given by  $i^s I_k$ .

To prove this claim, let  $c \in \overline{\mathcal{N}_F(\mathcal{A})}$ . Since  $i \in \mathcal{H}_R$  trivially,

$$\begin{aligned} \langle i, c \rangle &= \sum_{e \in \mathcal{E}} \text{trace}(i_e^T R_o c_e) = \sum_{e \in \mathcal{E}} i_e^s \text{trace}(R_o c_e) \\ &= \sum_{e \in \mathcal{E}} i_e^s \text{trace}(\bar{c}_e) = \sum_{e \in \mathcal{E}} i_e^s \sum_{l=1}^k \bar{c}_e^{(l,l)} \end{aligned}$$

where  $\bar{c}_e := R_o c_e \in \mathbb{R}^{k \times k}$  for every  $e \in \mathcal{E}$ , and  $\bar{c}_e^{(l,m)} \in \mathbb{R}$  represents the  $(l, m)^{\text{th}}$  scalar entry of the matrix  $\bar{c}_e$ . Hence,

$$\langle i, c \rangle = \sum_{e \in \mathcal{E}} \sum_{l=1}^k i_e^s \bar{c}_e^{(l,l)} = \sum_{l=1}^k \sum_{e \in \mathcal{E}} i_e^s \bar{c}_e^{(l,l)} = \sum_{l=1}^k \langle i^s, \bar{c}^{(l,l)} \rangle \quad (4.23)$$

where the  $k$  inner products on the right hand side are evaluated in the space  $\mathcal{H}_1$  defined for the scalar network  $(\mathcal{G}, 1)$ . Since  $c$  is a circulation,  $\mathcal{A}c = 0$ . Therefore

$$\sum_{e \in \mathcal{E}_p} a_{p,e} \bar{c}_e = R_o \sum_{e \in \mathcal{E}_p} a_{p,e} c_e = 0, \quad \forall p \in \mathcal{V},$$

where  $\mathcal{E}_p$  is the set of edges in  $\mathcal{G}$  that are incident on  $p$ , which shows that  $\bar{c}$  is also a circulation. Clearly, each scalar valued edge function  $\bar{c}^{(l,m)} : \mathcal{E} \rightarrow \mathbb{R}$  is also a circulation for the scalar electrical network  $(\mathcal{G}, 1)$ . It follows that  $\langle i^s, \bar{c}^{(l,l)} \rangle = 0$  for each  $l = 1, \dots, k$ . Hence, (4.23) implies

$$\langle i, c \rangle = 0$$

for every  $c \in \overline{\mathcal{N}(\mathcal{A})}$ , which is precisely the characterization of the current in  $\mathcal{H}_R$  stated in Corollary 4.9.1. This proves our claim that  $i^s I_k$  is the current in  $(\mathcal{G}, R_o)$ .

Therefore, the effective resistance is given by

$$R_{u,v}^{\text{eff}} = \sum_{e \in \mathcal{E}} i_e^T R_o i_e = R_o \sum_{e \in \mathcal{E}} (i_e^s)^2 = R_o r_{u,v}^{\text{eff}},$$

because of (4.6), which completes the proof. ■

*Proof of Proposition 4.6.1.* Pick an arbitrary  $j$ . Without loss of generality, assume that all the edges in  $\mathcal{E}_j$  are directed in the same way (cf. Remark 4.6.1). Construct a network  $(\mathcal{G}', R')$ , where  $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ , such that every set  $\mathcal{E}_j \subset \mathcal{E}$  is replaced by a single edge  $e_j \in \mathcal{E}'$ , and the edges resistances are assigned as

$$R_{e_j}'^{-1} := \sum_{e \in \mathcal{E}_j} R_e^{-1}. \quad (4.24)$$

The orientation of  $e_j$  is taken as that of the edges in  $\mathcal{E}_j$ . That is, if the edges in  $\mathcal{E}_j$  are incident on  $u$  and  $v$  and directed from  $u$  to  $v$ , then  $e_j := (u, v)$ .

Let  $i : \mathcal{E} \rightarrow \mathbb{R}^{k \times k}$  be the current in the network  $(\mathcal{G}, R)$  with intensity  $I_k$  from  $p$  to  $q$ , where  $p, q$  are two arbitrary nodes in  $\mathcal{G}$ . Assign a flow  $i' : \mathcal{E}' \rightarrow \mathbb{R}^{k \times k}$  in the graph  $\mathcal{G}'$  as

$$i'_{e_j} := \sum_{e \in \mathcal{E}_j} i_e, \quad j \in \mathbb{N}. \quad (4.25)$$

We will show first that  $i'$  is the current in  $(\mathcal{G}', R')$  with intensity  $I_k$  from  $p$  to  $q$ . It is easy to see that  $i'$  Kirchoff's current law (4.3). To check if Kirchoff's voltage law and Ohm's law are satisfied, pick an arbitrary  $j$ , and let the edges in  $\mathcal{E}_j$  be directed from  $u$  to  $v$  for some  $u, v \in \mathcal{V}$ . Let  $\ell$  be the number of edges in the set  $\mathcal{E}_j$ , and denote the edges in  $\mathcal{E}_j$  as  $e_1, e_2, \dots, e_\ell$ . Since the potential drop between  $u$  and  $v$  in the network  $(\mathcal{G}, R)$  is

$$V_u - V_v = R_{e_1} i_{e_1} = R_{e_2} i_{e_2} = \dots = R_{e_\ell} i_{e_\ell}, \quad (4.26)$$

we get

$$\begin{aligned}
R'_{ej} i'_{ej} &= \left( \sum_{e \in \mathcal{E}_j} R_e^{-1} \right)^{-1} (i_{e_1} + i_{e_2} + \dots i_{e_\ell}) \\
&= \left( \sum_{e \in \mathcal{E}_j} R_e^{-1} \right)^{-1} (i_{e_1} + R_{e_2}^{-1} R_{e_1} i_{e_1} + \dots + R_{e_\ell}^{-1} R_{e_1} i_{e_1}) \\
&= R_{e_1} i_{e_1} = V_u - V_v.
\end{aligned}$$

This shows that  $i'$  is a current in the network  $(\mathcal{G}', R')$  with node potential function  $V$  - the same as in  $(\mathcal{G}, R)$ . Therefore, the potential drop between  $p$  and  $q$ , which is the effective resistance between them, is the same in the two networks. The same argument applies to all node pairs, which proves the result. ■

# Chapter 5

## Error scaling laws

### 5.1 Introduction

In this chapter we answer the error-scaling question on estimation with relative measurements, that was raised at the beginning of Chapter 1. We want to examine how the minimum possible estimation error of a node variable  $x_u$  varies with the node's distance from the reference node  $o$  in a large measurement graph, and how this variation is affected by the structure of the measurement graph.

As discussed in Chapter 1 and again in Section 4.1, the error scaling question is important for large measurement graphs. Therefore we consider the limiting case of *infinite* graphs, in which the number of variables and available measurements are countably infinite. It was shown in the preceding chapter that the optimal estimation error of a node variable in a large but finite subgraph of the infinite graph can be made arbitrarily close to the error in the infinite graph, by making the finite graph sufficiently large. Intuitively, for a fixed node, as the graph becomes larger and larger, it appears to extend to infinity in all directions from the point of

view of the node. Therefore, as long as we are interested in the error covariance of a node that is sufficiently inside a large graph (i.e., not too close to the boundary), conclusions drawn for infinite graphs are applicable to large but finite graphs. The advantage is that analyzing infinite graphs is often easier, since boundary conditions are not as important in an infinite graph as it is in a finite graph. In addition, the minimum estimation error achievable in an arbitrarily large graph can be characterized by the limiting BLUE covariance in an infinite graph that was defined in the previous chapter. For this reason, *in this chapter we examine the scaling laws for the BLU estimation error covariance in infinite measurement graphs.*

The structure of a measurement graph has a direct bearing on how the estimation error of a node variable varies with its distance from the reference. When the measurement graph is a tree, there is a single path between the  $u^{\text{th}}$  node and the reference node and one can show that the covariance matrix of the estimation error is the sum of the covariance matrices associated with this path. Thus, for trees, the variance of the optimal estimation error of  $x_u$  grows linearly with the distance of node  $u$  from the reference node. It turns out that for graphs “denser” than trees, with multiple paths between pairs of nodes, the variance of the optimal estimation error can grow less than linearly with distance.

However, the notion of denseness of a graph is not easy to define. In classical graph-theoretic terminology, a graph with  $n$  vertices is called dense if its average node degree is of order  $n$ , and is called sparse if its average node degree is a constant independent of  $n$  [110]. Recall that the degree of a node refers to the number of edges that are incident on it. An edge  $(u, v)$  is said to be incident on the nodes  $u$  and  $v$ . In the sensor network literature that examines the accuracy of

location estimation from range measurement, graph density is recognized to affect estimation accuracy; although graph density is measured by the average number of nodes per unit area of a deployed network [15, 16]. However, none of these measures determines how the estimation error scales with the size of the graph, as we will see through examples in Section 5.3.3.

A notion of graph denseness and sparseness that is useful in predicting error scaling laws can be developed by examining the relationship between the graph and a lattice. The  $d$ -dimensional square lattice  $\mathbf{Z}_d$  is defined as a graph with a node in every point in  $\mathbb{R}^d$  with integer coordinates and an edge between every pair of nodes that have a Euclidean distance 1 between them (see Figure 5.3 for examples). The error scaling laws for a lattice measurement graph can be determined analytically by exploiting the symmetry of the lattice.

When the graph is not a lattice, it can still be compared to a lattice. Intuitively, if a graph, after some bounded perturbation in its node and edge set, looks approximately like a  $d$ -dimensional lattice, the graph is as dense as a lattice. In that case the error covariance in the lattice can be used to bound the error covariance in the graph. It also turns out that the graphs that can be compared to lattices in this manner are realistic models of sensor networks obtained by placing nodes in a geographical area in an ad-hoc fashion. Thus, the error scaling laws obtained for these graphs turn out to be quite useful in design and deployment of realistic sensor networks.

*Chapter organization:* After outlining the contributions of this chapter in Section 5.2, we pose the error scaling problem precisely in Section 5.3. This section also contains the main results of this chapter in the form of theorems and lemmas. Although the characterization of graphs according to how fast the errors

grow with distance are provided in the beginning of Section 5.3, in Section 5.4 we return to these graphs for a more extensive discussion of their properties. Section 5.5 starts with scaling laws for lattices and a close relative of theirs, and hints at how these can be extended to a much broader class by thinking of graphs as coarse approximation of Euclidean spaces, and ends by providing the formal proof of the scaling laws that were stated earlier in Section 5.3. The chapter ends with a discussion of open problems in Section 5.6.

## 5.2 Contributions and prior work.

The results established in this chapter and their implications are summarized below:

1. We derive a classification of graphs, dense and sparse graphs in  $\mathbb{R}^d$ ,  $d = 1, 2, 3$ , that determines the rate at which the limiting BLUE covariance of a node variable changes with the node's distance from the reference. For dense graphs, upper bounds on the growth rate of the error, and for sparse graphs, lower bounds on the estimation error growth, are obtained. In particular, when a graph is dense in 1D, 2D, and 3D, respectively, the error covariance of a node is upper bounded by a linear, logarithmic, and bounded function of its distance from the reference. On the other hand, when a graph is sparse in 1D, 2D, and 3D respectively, the error covariance of a node is lower bounded by a linear, logarithmic, and bounded function of its distance from the reference.
2. The error scaling laws derived in this chapter puts an algorithm-independent limit to the estimation accuracy achievable in large networks, since no esti-

mation algorithm can achieve higher accuracy than the optimal estimator. For this reason, the bounds and the associated graph classification can be useful in performance analysis, design, and deployment of large networks. For example, when a graph is sparse in 1D, the optimal estimation error covariance grows at least linearly with the distance from the reference. Therefore the estimation accuracy will be necessarily poor in 1D sparse graphs. Recognizing the sparseness of the graph will help the user to realize that, either high estimation accuracy cannot be achieved, or more reference nodes need to be introduced. On the other hand, when a graph is dense in 3D, the optimal estimation error of every node variable remains below a constant, even for nodes that are arbitrary far away from the reference node. So accurate estimation is possible in 3D dense graphs. One can therefore try to deploy networks that satisfy denseness properties so that guarantees on the estimation error can be provided a-priori.

3. We show that graphs obtained by placing nodes in a geographical area in an ad-hoc fashion are likely to fall into one of the classes of graphs identified here. Since we now know which structural properties are beneficial for accurate estimation, and such structures are achievable by realistic sensor networks, we can strive to achieve those structures in deploying a network.
4. The results described in this chapter expose certain misconceptions that exist in the sensor network literature about the relationship between graph structure and estimation error. In Section 5.3.3, we provide examples that show the inadequacy of the usual measures of graph denseness, such as node degree, in determining scaling laws of the estimation error.

The material presented in this chapter was published in a preliminary form in [111].

*Prior work:* Although the problem of estimation from relative measurements appears in several sensor and ad-hoc network applications, there has been no systematic study on the effect of network structure on the achievable estimation error. Among all the possible applications discussed in Section 2.1, localization and time-synchronization in sensor networks have attracted the most interest from the research community.

However, the the majority of the existing literature on localization is concerned with developing algorithms for estimating locations from relative range measurements alone, as discussed in Section 2.1.1. Notwithstanding this major difference of the problem formulation, the few papers that have attempted to examine the effect of various parameters on the accuracy of localization from range measurements, such as [15, 16, 37, 112], fails to provide a clear answer to the question of how network structure affects estimation accuracy. Most of the papers concluded that high node degree is beneficial to estimation accuracy [15, 37].

There is a substantial body of literature on time-synchronization from relative clock offset measurements, but the typical estimation algorithms do not attempt to compute the optimal estimates. To the best of our knowledge, optimal clock offset estimation from relative measurements was examined for the first time by Karp *et al.* [73], and thereafter by Barooah *et al.* [65], Barooah and Hespanha [67] and then by [113]. However, the focus of these papers was distributed computation of the estimates and not the examination of network structure's effect on achievable estimation accuracy.

### 5.3 Problem statement and main results

Recall that in this chapter we consider the problem of estimating a countably infinite vector-valued variables  $x_u \in \mathbb{R}^k$ ,  $u \in \mathcal{V} := \{1, 2, \dots\}$ , from noisy relative measurements of the form:

$$\zeta_{u,v} = x_u - x_v + \epsilon_{u,v}, \quad (u, v) \in \mathcal{E}$$

where  $\epsilon_{u,v}$  denotes a zero-mean measurement noise and  $\mathcal{E}$  is the set of ordered pairs  $(u, v)$  for which relative measurements are available. We assume that the value of a particular *reference variable*  $x_o$  is known, and without loss of generality we take  $x_o = 0$ . The node set  $\mathcal{V}$  and the edge set  $\mathcal{E}$  together define a directed measurement graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ .

The accuracy of a node variable's estimate, measured in terms of the covariance of the estimation error, depends on the graph  $\mathcal{G}$  as well as the measurement errors. The covariance matrix of the error  $\epsilon_{u,v}$  in the measurement  $\zeta_{u,v}$  is denoted by  $P_{u,v}$ , i.e.,  $P_{u,v} := \mathbb{E}[\epsilon_{u,v}\epsilon_{u,v}^T]$ . The measurement errors on different edges are uncorrelated, i.e., for every pair of distinct edges  $e, \bar{e} \in \mathcal{E}$ ,  $\mathbb{E}[\epsilon_e\epsilon_{\bar{e}}^T] = 0$ . The estimation problem is now formulated in terms of a *network*  $(\mathcal{G}, P)$  where  $P : \mathcal{E} \rightarrow \mathbb{S}^{k+}$  is a function that assigns to each edge  $(u, v) \in \mathcal{E}$  the covariance matrix  $P_{u,v}$  of the measurement error associated with the edge  $(u, v)$  in the measurement graph  $\mathcal{G}$ .

The problem is to determine how the BLUE covariance  $\Sigma_{u,o}$  scales as a function of the distance of node  $u$  from the reference  $o$ , and how this scaling law depends on the structure of the measurement graph  $\mathcal{G}$ . In view of the electrical analogy established in the previous chapter, specifically Theorem 4.5.1, the question can be equivalently posed in terms of the generalized effective resistance. In the sequel,

we only deal with the effective resistance. Now we define a classification of graphs for which the scaling laws for the effective resistance, and thereby of the optimal estimation error, can be determined.

*Remark 5.3.1 (Assumptions).* Recall from Chapter 4 that a measurement network is assumed to satisfy the conditions of Assumption 4.2.1, which stipulates that the measurement graph is weakly connected, it has a finite maximum node degree, and the edge covariances are uniformly bounded. In this chapter, we impose the additional assumption that there are no parallel edges in the measurement graph. This assumption is not restrictive since parallel measurement edges can be combined into a single one with an appropriate covariance, which preserves the BLUE covariances, which follows from Proposition 4.6.1 and the analogy between BLU covariance and effective resistance.  $\square$

### 5.3.1 Graph denseness and sparseness

We start with graph drawing, which will allow us to define dense and sparse graphs.

#### 5.3.1.1 Graph drawing

The drawing of a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  in a  $d$ -dimensional Euclidean space is obtained by mapping the nodes into points in  $\mathbb{R}^d$  by a *drawing function*  $f : \mathcal{V} \rightarrow \mathbb{R}^d$ . A drawing is also called a *representation* of a graph [60]. For a particular drawing  $f$  of a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , given two nodes  $u, v \in \mathcal{V}$  the *Euclidean distance between  $u$  and  $v$  induced by the drawing  $f : \mathcal{V} \rightarrow \mathbb{R}^d$*  is defined by

$$d_f(u, v) := \|f(v) - f(u)\|,$$

where  $\|\cdot\|$  denoted the usual Euclidean norm in  $\mathbb{R}^d$ . It is important to emphasize that the definition of drawing does not require edges to not intersect and therefore every graph has a drawing in every Euclidean space. In fact, every graph has an infinite number of drawings in every Euclidean space. However, some drawings are more useful than others in clarifying the relationship between the graph and the Euclidean space in which it is drawn. It is this relationship that is the key to defining an appropriate measure of graph denseness and sparseness.

For a particular drawing  $f$  and induced Euclidean distance  $d_f$  of a graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , four parameters can be used to characterize graph denseness and sparseness. The *minimum node distance*, denoted by  $s$ , is defined as the minimum Euclidean distance between the drawing of two nodes

$$s := \inf_{\substack{u, v \in \mathcal{V} \\ v \neq u}} d_f(u, v).$$

The *maximum connected range*, denoted by  $r$ , is defined as the Euclidean length of the drawing of the longest edge

$$r := \sup_{(u, v) \in \mathcal{E}} d_f(u, v).$$

The *maximum uncovered diameter*, denoted by  $\gamma$ , is defined as the diameter of the largest open ball that can be placed in  $\mathbb{R}^d$  such that it does not enclose the drawing of a node

$$\gamma := \sup \left\{ \delta : \exists \mathcal{B}_\delta \text{ s.t. } f(u) \notin \mathcal{B}_\delta, \forall u \in \mathcal{V} \right\},$$

where the existential quantification spans over the balls  $\mathcal{B}_\delta$  in  $\mathbb{R}^d$  with diameter  $\delta$  and centered at arbitrary points. Finally, the *asymptotic distance ratio*, denoted by  $\rho$ , is defined as

$$\rho := \lim_{n \rightarrow \infty} \inf \left\{ \frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} : u, v \in \mathcal{V} \text{ and } d_{\mathcal{G}}(u, v) \geq n \right\},$$

where  $d_G(u, v)$  denotes the graphical distance between  $u$  and  $v$  in the graph  $\mathcal{G}$ . Essentially  $\rho$  provides a lower bound for the ratio between the Euclidean and the graphical distance for nodes that are far apart. The asymptotic distance ratio can be thought of as an inverse of the *stretch* for geometric graphs, which is a well-studied concept for finite graphs [114].

If the asymptotic distance ratio  $\rho$  is positive for the drawing of graph, or its maximum connected range  $r$  is finite, it says something about the relation between the graphical distances between nodes the Euclidean distance between their drawings, which is stated in the next result. The proof is provided in Section 5.7.

**Lemma 5.3.1 ( $\rho$  vs. linear growth).** *The following two statements are equivalent:*

1. *The asymptotic distance ratio  $\rho$  is strictly positive.*
2. *There exist constants  $\alpha > 0, \beta > 0$  for which*

$$d_G(u, v) \leq \alpha d_f(u, v) + \beta, \quad \forall u, v \in \mathcal{V}.$$

*Similarly, the following statements are equivalent:*

1. *The maximum connected range  $r$  is finite.*
2. *There exist real constants  $\alpha > 0, \beta \geq 0$  for which*

$$d_f(u, v) \leq \alpha d_G(u, v) + \beta, \quad \forall u, v \in \mathcal{V}. \quad \square$$

In other words, when  $\rho > 0$ , small Euclidean distance in the drawing implies small graphical distance in the drawing. On the other hand, when  $r < \infty$ , small graphical distance implies small Euclidean distance.

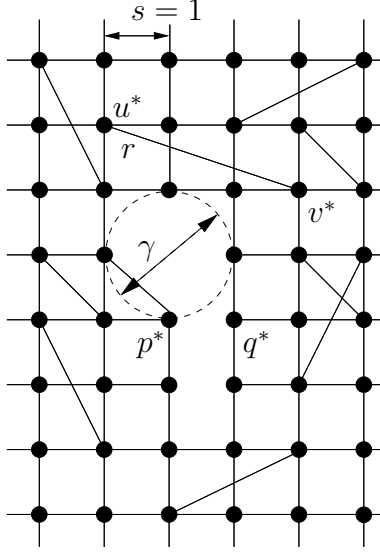


Figure 5.1. A drawing of a graph in 2D Euclidean space, and the corresponding denseness and sparseness parameters. Since the minimal distance between any two nodes is 1, so the minimum node distance is  $s = 1$ . Since the longest edge is between  $u^*$  and  $v^*$ , of length  $\sqrt{10}$ , the maximum connected range is  $r = \sqrt{10}$ . The diameter of the largest ball that can fit inside the drawing without enclosing any node is 2, the maximum uncovered diameter is thus  $\gamma = 2$ . The minimal ratio between the Euclidean and graphical distance of a pair of nodes is achieved by the pair  $p^*, q^*$ , hence the asymptotic distance ratio is  $\rho = d_f(p^*, q^*)/d_G(p^*, q^*) = 1/5$ .

### 5.3.1.2 Dense and Sparse Graphs

The drawing of a graph for which the maximum uncovered diameter is finite ( $\gamma < \infty$ ) and the asymptotic distance ratio is positive ( $\rho > 0$ ) is called a *dense drawing*. We say that a graph  $\mathcal{G}$  is *dense in  $\mathbb{R}^d$*  if there exists a dense drawing of the graph in  $\mathbb{R}^d$ . Intuitively, these drawing are dense in the sense that the nodes can cover  $\mathbb{R}^d$  without leaving large holes between them and still having sufficiently many edges so that a small Euclidean distance between two nodes in the drawing

guarantees a small graphical distance between them.

A graph drawing for which the minimum node distance is positive ( $s > 0$ ) and the maximum connected range is finite ( $r < \infty$ ) is called a *civilized drawing* [2]. A graph  $\mathcal{G}$  is said to be *sparse in  $\mathbb{R}^d$*  if there exists a civilized drawing of it in  $\mathbb{R}^d$ . Intuitively, these drawings are sparse in the sense that one can keep the edges with finite lengths without cramping all nodes on top of each other.

*Remark 5.3.2 (historical note).* A graph that is sparse in  $\mathbb{R}^d$  is a graph that can be drawn in a civilized manner in  $\mathbb{R}^d$ ; where the notion of “a graph that can be drawing a civilized manner” was introduced by Doyle and Snell [2] in connection with random walks. In this dissertation we refer to such graphs as sparse graphs since they are the antitheses of dense graphs.  $\square$

A graph can be both dense and sparse in the same dimension. For example, consider the  $d$ -dimensional square lattice  $\mathbf{Z}_d$ , which is defined as a graph with a node in every point in  $\mathbb{R}^d$  with integer coordinates and an edge between every pair of nodes that have a Euclidean distance 1 between them (see Figure 5.3 for examples). We can conclude from the definition of a lattice (which defines a drawing as well) that the  $d$ -dimensional lattice is both sparse and dense in  $\mathbb{R}^d$ . However, there is no civilized drawing of the  $d$ -dimensional lattice in  $\mathbb{R}^{\underline{d}}$  for any  $\underline{d} < d$ . Moreover, there is no dense drawing of the  $d$ -dimensional lattice in  $\mathbb{R}^{\bar{d}}$  for every  $\bar{d} > d$ . This means, for example, that the 3D lattice is not sparse in 2D and is not dense in 4D. In general, a graph being dense in a particular dimension puts a restriction on which dimensions it can be sparse in. The next result, proved in Section 5.4.2, states this precisely.

**Lemma 5.3.2.** *If a graph is dense in  $\mathbb{R}^d$  for some  $d \geq 1$ , it is not sparse in  $\mathbb{R}^{\underline{d}}$  for every  $\underline{d} < d$ .*  $\square$




Euclidean space	Covariance matrix $\Sigma_{u,o}$ of the estimation error of $x_u$ in a <i>sparse graph</i> with a civilized drawing $f'_d$	Covariance matrix $\Sigma_{u,o}$ of the estimation error of $x_u$ in a <i>dense graph</i> with a dense drawing $f_d$
 $\mathbb{R}$	$\Sigma_{u,o}(\mathcal{G}) = \Omega\left(d_{f'_1}(u, o)\right)$	$\Sigma_{u,o}(\mathcal{G}) = \mathcal{O}\left(d_{f_1}(u, o)\right)$
 $\mathbb{R}^2$	$\Sigma_{u,o}(\mathcal{G}) = \Omega\left(\log d_{f'_2}(u, o)\right)$	$\Sigma_{u,o}(\mathcal{G}) = \mathcal{O}\left(\log d_{f_2}(u, o)\right)$
 $\mathbb{R}^3$	$\Sigma_{u,o}(\mathcal{G}) = \Omega(1)$	$\Sigma_{u,o}(\mathcal{G}) = \mathcal{O}(1)$

Table 5.1. Covariance matrix  $\Sigma_{u,o}$  of  $x_u$ 's optimal estimate for graphs that are dense or sparse in  $\mathbb{R}^d$ . In the table,  $d_{f_d}(u, o)$  denotes the Euclidean distance between node  $u$  and the reference node  $o$  for any drawing  $f_d : \mathcal{V} \rightarrow \mathbb{R}^d$  that establishes the graph's denseness in the Euclidean space  $\mathbb{R}^d$ , and  $d_{f'_d}(u, o)$  denotes the Euclidean distance in any drawing  $f'_d$  that establishes the graph's sparseness in  $\mathbb{R}^d$ .

### 5.3.2 Error scaling laws

The concepts of dense and sparse graphs allow one to characterize precisely how the BLUE error covariance  $\Sigma_{u,o}$  grows with the distance of the node  $u$  from the reference  $o$  in infinite measurement graphs. The following theorem, proved in Section 5.5.3, establishes the scaling laws for the BLUE error covariances in dense and sparse graphs. The theorem is an answer to the error scaling question raised in Chapter 1.

**Theorem 5.3.1 (Error Scaling Laws).** *Consider a measurement network  $(\mathcal{G}, P)$  that satisfies Assumption 4.2.1, such that the graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  has a reference node  $o \in \mathcal{V}$ . Then, the limiting BLUE error covariance  $\Sigma_{u,o}$  for every node  $u \in \mathcal{V} \setminus \{o\}$  obeys the scaling laws shown in Table 5.1.*  $\square$

In Table 5.1, the usual asymptotic notations  $\Omega(\cdot)$  and  $\mathcal{O}(\cdot)$  are used with matrix valued functions in the following way. For a matrix-valued function  $g : \mathbb{R} \rightarrow \mathbb{R}^{k \times k}$  and a scalar-valued function  $p : \mathbb{R} \rightarrow \mathbb{R}$ , the notation  $g(x) = \mathcal{O}(p(x))$  means that there exists a positive constant  $x_o$  and a constant matrix  $A \in \mathbb{S}^{k+}$  such that  $g(x) \leq Ap(x)$  for all  $x > x_o$ . Similarly,  $g(x) = \Omega(p(x))$  means there exists a positive constant  $x_o$  and a constant matrix  $B \in \mathbb{S}^{k+}$  such that  $g(x) \geq Bp(x)$  for all  $x > x_o$ . Recall that  $\mathbb{S}^{k+}$  is the set of all  $k \times k$  symmetric positive definite matrices.

If a graph is both sparse and dense in a particular Euclidean space  $\mathbb{R}^d$ , the asymptotic upper and lower bounds for the error covariance is the same. The effective resistance in such a graph grows with distance in the same rate as it grows in the  $d$ -D lattice. Intuitively, such a graph behaves approximately like a lattice.

Since a graph can be dense and sparse in multiple dimensions, one may wonder if it is possible to encounter the situation in which a graph is dense in  $\mathbb{R}^2$  *as well as* sparse in  $\mathbb{R}$ , which will lead to a logarithmic upper bound in one drawing and a linear lower bound in another drawing. Such an undesirable situation, however, is precluded by Lemma 5.3.2.

### 5.3.3 Counterexamples to conventional wisdom

It was pointed out in Section 5.1 that typically, the average node degree of a graph or the number of nodes and edges per unit area of a deployed network is used as a measure of graph denseness. However, these measures do not predict error scaling laws. The three graphs in Figure 5.2 offer an example of the inadequacy of node degree as a measure of denseness. It shows a 3-fuzz of the 1D lattice (see

Section 5.4 for the definition of a lattices and fuzzes), a triangular lattice, and a 3-dimensional lattice. It can be verified from the definitions in Section 5.3.1.2 that the 3-fuzz of the 1D lattice is both dense and sparse in  $\mathbb{R}$ , the triangular lattice is dense and sparse in  $\mathbb{R}^2$ , and the 3D lattice is dense and sparse in  $\mathbb{R}^3$ . Thus, it follows from Theorem 5.3.1 that the BLU estimation error scales linearly with distance in the 3-fuzz of the of the 1D lattice, logarithmically with distance in the triangular lattice, and is uniformly bounded with respect to distance in the 3D lattice, even though every node in each of these graphs has the same degree, namely six.

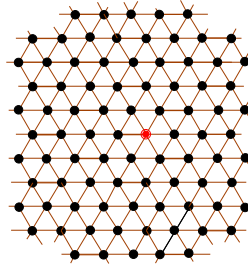
## 5.4 Dense and sparse graphs

The dense and sparse graphs defined in Section 5.3.1.2 have a special relationship with lattices and a close relative of lattices - called lattice fuzzes – in terms of embedding.

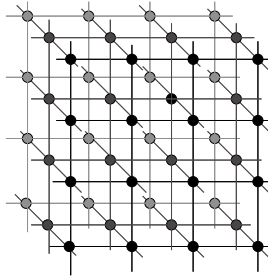
Recall that the  $d$ -dimensional square lattice  $\mathbf{Z}_d$  is defined as a graph with a node in every point in  $\mathbb{R}^d$  with integer coordinates and an edge between every pair of nodes that have a Euclidean distance 1 between them (see Figure 5.3 for examples). The  $h$ -fuzz of a graph  $\mathcal{G}$ , introduced by Doyle and Snell [2] is a graph with the same set of nodes as  $\mathcal{G}$  but with a larger set of edges. Given a graph  $\mathcal{G}$  and a positive integer  $h$ , the  $h$ -fuzz of  $\mathcal{G}$ , denoted by  $\mathcal{G}^{(h)}$ , is the graph that has an edge between two nodes  $u$  and  $v$  whenever the graphical distance between them in  $\mathcal{G}$  is less than or equal to  $h$ . The graphical distance  $d_G$  is evaluated without regards to edge directions. In view of remark 4.6.1, the edge directions are irrelevant so long as we are interested only in the effective resistance.



(a) A 3-fuzz of a 1D lattice



(b) A triangular lattice



(c) A 3D lattice

Figure 5.2. Three measurement graphs that show vastly different scaling laws of the estimation error, whereas each has the same node degree for every node. Furthermore, they are all “sparse” according to traditional graph-theoretic terminology (see the discussion on graph denseness in Section 5.1).

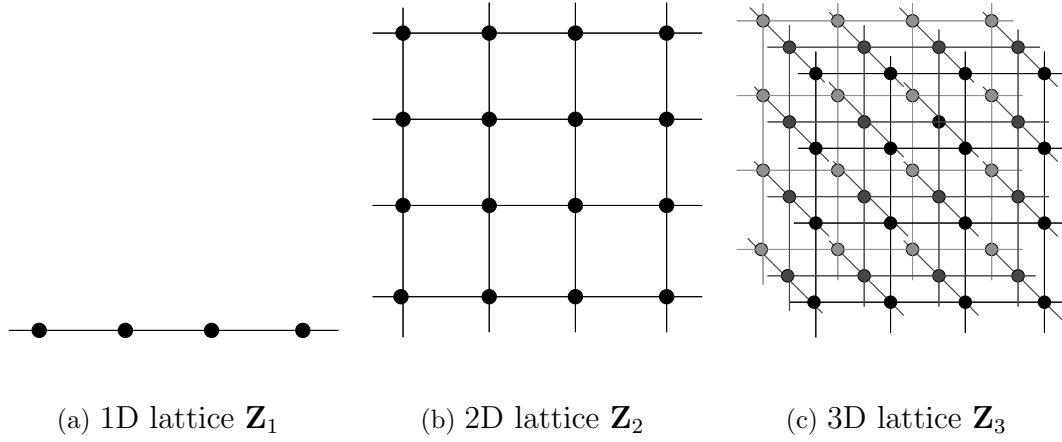


Figure 5.3. Lattices.

Recalling the definition of embedding from Section 4.6.1, and noting that there are no parallel edges by assumption (see Remark 5.3.1), we note that  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$  if it can be made a subgraph of  $\bar{\mathcal{G}}$  by relabeling its nodes and disregarding edge directions. Recall also that we write  $\mathcal{G} \subset \bar{\mathcal{G}}$  to denote  $\mathcal{G}$  can be embedded in  $\bar{\mathcal{G}}$ .

### 5.4.1 Relationship with lattices

The next theorem shows that fuzzes of dense graphs can embed lattices. The proof of the result is provided in Section 5.7. We use  $d_{\mathbf{Z}_d}(\cdot)$  to denote the graphical distance in  $\mathbf{Z}_d$  and  $d_f(\cdot)$  to denote the Euclidean distance in  $\mathbb{R}^d$  induced by a drawing  $f$ .

**Theorem 5.4.1 (Dense Embedding).** *A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is dense in  $\mathbb{R}^d$  if and only if there exists finite, positive integers  $h$  and  $c$  such that the following conditions are satisfied*

1.  $\mathcal{G}^{(h)} \supset \mathbf{Z}_d$ , and,

2. if  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$  is an embedding of  $\mathbf{Z}_d$  into  $\mathcal{G}^{(h)}$ , then,  $\forall u \in \mathcal{V}, \exists \bar{u} \in \eta(\mathcal{V}_{\mathbf{Z}_d}) \subseteq \mathcal{V}$  such that  $d_{\mathcal{G}}(u, \bar{u}) \leq c$ .

Moreover, if  $f : \mathcal{V} \rightarrow \mathbb{R}^d$  is a dense drawing of  $\mathcal{G}$  in  $\mathbb{R}^d$  and  $\eta$  is an embedding function that satisfies condition (2) above, then the following is also true:  $\forall u, v \in \mathcal{V}$ , we can find  $u_z, v_z \in \mathcal{V}_{\mathbf{Z}_d}$  which satisfies

$$\begin{aligned} d_{\mathcal{G}}(u, \eta(u_z)) &\leq c, & d_{\mathcal{G}}(v, \eta(v_z)) &\leq c \\ d_{\mathbf{Z}_d}(u_z, v_z) &\leq 4d + \frac{\sqrt{d}}{\gamma} d_f(u, v) \end{aligned} \tag{5.1}$$

where  $\gamma$  is the maximum uncovered diameter of the  $f$ -drawing of  $\mathcal{G}$ .  $\square$

In other words,  $\mathcal{G}$  is dense in  $\mathbb{R}^d$  if and only if (i) the  $d$ -dimensional lattice can be embedded in an  $h$ -fuzz of  $\mathcal{G}$  for some positive integer  $h$  and (ii) every node of  $\mathcal{G}$  that is not the image of a node in  $\mathbf{Z}_d$  is at a uniformly bounded graphical distance from a node that is the image of a node in  $\mathbf{Z}_d$ . The significance of (5.1) is that not only can we find for every node in  $\mathcal{G}$  a close-by node that has a pre-image in the lattice, but also these close-by nodes can be so chosen that if the Euclidean distance between a pair of nodes  $u$  and  $v$  in a dense drawing of the graph is small, then the graphical distance in the lattice between the pre-images of their close-by nodes is small as well.

The next theorem shows that a graph that is sparse in  $\mathbb{R}^d$  can be embedded in a fuzz of the  $d$ -dimensional lattice. The proof of the theorem is provided in Section 5.7.

**Theorem 5.4.2 (Sparse Embedding).** *A graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is sparse in  $\mathbb{R}^d$  if and only if there exists a positive integer  $h$  such that  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$ . Moreover, if  $f : \mathcal{V} \rightarrow \mathbb{R}^d$  is a civilized drawing of  $\mathcal{G}$  in  $\mathbb{R}^d$ , then there exists an embedding*

$\eta : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$  so that  $\forall u, v \in \mathcal{V}$ ,

$$d_{\mathbf{Z}_d}(\eta(u), \eta(v)) \geq \sqrt{d} \left( \frac{1}{s} d_f(u, v) - 2 \right). \quad (5.2)$$

, where  $s$  is the minimum node-distance of the  $f$ -drawing of  $\mathcal{G}$ .  $\square$

In other words,  $\mathcal{G}$  is sparse in  $\mathbb{R}^d$  if and only if  $\mathcal{G}$  can be embedded in an  $h$ -fuzz of a  $d$ -dimensional lattice. The significance of (5.2) is that if the Euclidean distance between a pair of nodes in a civilized drawing of the graph is large, the graphical distance in the lattice between their corresponding nodes is also large.

The first statement of the theorem is essentially taken from [2], where it was proved that if a graph can be drawn in a civilized manner in  $\mathbb{R}^d$ , then it can be embedded in a  $h$ -fuzz of a  $d$ -lattice, where  $h$  depends only on  $s$  and  $r$ . A careful examination of the proof reveals that it is not only sufficient but also a necessary condition for embedding in lattice fuzzes.

### 5.4.2 Checking denseness and sparseness

To show a graph is dense (or sparse) in a particular dimension, one has to find a drawing in that dimension with the appropriate properties. Dense and sparse graphs occur readily with realistic “communication range” models, in which nodes are deployed in an Euclidean space – perhaps randomly – and two nodes form an edge between them if they are within range of each other [115]. A widely studied class of such graphs that is also highly relevant for engineering applications is the random geometric graph [109]. For such graphs, a *natural drawing* is obtained by mapping the nodes to their physical locations in the Euclidean space they are deployed in. We can show using the natural drawing, that, a graph generated by placing a countable number of nodes in  $\mathbb{R}^d$ , so that the maximum uncovered

diameter  $\gamma$  of its natural drawing is finite, and every pair of nodes whose Euclidean distance in the natural drawing is less than  $2\delta$  has an edge between them, is dense in  $\mathbb{R}^d$ . Such communication-range models also yield sparse graph quite easily, since the condition of finite maximum connected range is satisfied by construction.

On the other hand, to show that a graph is *not* dense (or not sparse) in a particular dimension is harder since one has to show that no drawing is possible that has the required properties. Typically, this can be done by showing that the existence of a dense (or sparse) drawing leads to a contradiction. An application of this technique leads to the following result.

**Lemma 5.4.1.**    1. *The  $d$ -dimensional lattice  $\mathbf{Z}_d$  is not sparse in  $\mathbb{R}^{\underline{d}}$  for every  $\underline{d} < d$ , and it is not dense in  $\mathbb{R}^{\bar{d}}$  for every  $\bar{d} > d$ .*

2. *A regular<sup>1</sup> infinite tree is not dense or sparse in any dimension.*  $\square$

The first statement of the lemma is provided in Section 5.7. The proof of the second statement is not provided since the method of the proof is similar.

We are now ready to prove Lemma 5.3.2.

*Proof of Lemma 5.3.2.* To prove the result by contradiction, suppose that a graph  $\mathcal{G}$  is dense in  $\mathbb{R}^d$  as well as sparse in  $\mathbb{R}^{\bar{d}}$ , where  $\bar{d} < d$ . It follows from Theorems 5.4.1 and 5.4.2 that there exist positive integers  $\ell, p$  such that  $\mathbf{Z}_d \subset \mathcal{G}^{(\ell)}$  and  $\mathcal{G} \subset \mathbf{Z}_{\bar{d}}^{(p)}$ . It is straightforward to verify the following facts:

1. for every pair of graphs  $\mathcal{G}, \bar{\mathcal{G}}$  that do not have any parallel edges,  $\mathcal{G} \subset \bar{\mathcal{G}} \Rightarrow \mathcal{G}^{(l)} \subset \bar{\mathcal{G}}^{(l)}$  for every positive integer  $l$ .

---

<sup>1</sup>A graph is called regular if the degree of every node in the graph is the same.

2. for an arbitrary graph  $\mathcal{G}$  without parallel edges, and two positive integers  $\ell, p$ , we have  $(\mathcal{G}^{(p)})^{(\ell)} = \mathcal{G}^{(p\ell)}$ .

It follows that  $\mathbf{Z}_d \subset \mathbf{Z}_{\bar{d}}^{(\ell p)}$ , which means, from sparse embedding Theorem 5.4.2, that a  $d$ -dimensional lattice is sparse in  $\mathbb{R}^{\bar{d}}$ . This is a contradiction because of Lemma 5.4.1, which completes the proof.  $\blacksquare$

## 5.5 Establishing the error scaling laws

Here we briefly outline the approach by which the error scaling laws stated in Theorem 5.3.1 are obtained, and how the definitions of dense and sparse graphs allow one to obtain those results. The key idea is to embed the measurement graph in a “nice” looking graph such that the effective resistances in the nice graph can be computed. Application of Rayleigh’s monotonicity law then tells us that the effective resistance in the nice graph is a lower bound on the effective resistance in the measurement graph, and from the electrical analogy we get a lower bound on the BLUE covariance. Similarly, when we can embed a nice graph in the measurement graph, we get an upper bound on the BLUE covariances.

The nice graphs that we use for the embedding are lattices and their fuzzes. The effective resistance in lattices and their fuzzes can be analytically computed because of the symmetry in their structure. It will be shown shortly that the effective resistance in 1D, 2D and 3D lattices is a linear, logarithmic, and bounded function of distance. Since dense graphs can embed lattice-like graphs, namely fuzzes of lattices, we can show that the effective resistance in graphs that are dense in 1D, 2D, and 3D grow as a linear, logarithmic, and bounded function of distance as well. A similar story unfolds for sparse graphs.

### 5.5.1 Effective resistance for lattices and fuzzes

An  $h$ -fuzz will clearly have lower effective resistance than the original graph because of Rayleigh's Monotonicity Law, but it is lower only by a constant factor. The following result states this feature of fuzzes, which is a straightforward extension to the generalized case of a result about scalar effective resistance established by Doyle [116]. Since the proof in [116] uses terminology of random walks, we still include a proof in Section 5.5.3.

**Lemma 5.5.1.** *Let  $(\mathcal{G}, R_o)$  be a generalized electrical network with graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  satisfying Assumption 4.2.1, with a constant generalized resistance  $R_o \in \mathbb{S}^{k+}$  on its every edge. Let  $(\mathcal{G}^{(h)}, R_o)$  be the electrical network similarly constructed on  $\mathcal{G}^{(h)}$ , the  $h$ -fuzz of  $\mathcal{G}$ . For every pair of nodes  $u$  and  $v$  in  $\mathcal{V}$ ,*

$$\alpha R_{u,v}^{\text{eff}}(\mathcal{G}) \leq R_{u,v}^{\text{eff}}(\mathcal{G}^{(h)}) \leq R_{u,v}^{\text{eff}}(\mathcal{G}),$$

where  $R_{u,v}^{\text{eff}}(\cdot)$  is the effective resistance in the network  $(\cdot, R_o)$  and  $\alpha \in (0, 1]$  is a positive constant that does not depend on  $u$  and  $v$ .  $\square$

The following lemma establishes effective resistances in  $d$ -dimensional lattices and their fuzzes. Note that infinite generalized networks constructed by assigning constant matrix-resistances on every edge of a lattice or a  $h$ -fuzz of it satisfies Assumption 4.2.1, and therefore results in Chapter 4 guarantees that the effective resistances in infinite lattice networks are well defined.

**Lemma 5.5.2.** *Consider the electrical network  $(\mathbf{Z}_d^h, R_o)$  with a constant generalized resistance  $R_o \in \mathbb{S}^{k+}$  at every edge of the  $h$ -fuzz of the  $d$ -dimensional square lattice  $\mathbf{Z}_d$ , where  $h$  is an integer. The generalized effective resistance  $R_{u,v}^{\text{eff}}$  between two nodes  $u$  and  $v$  in the electrical network  $(\mathbf{Z}_d^h, R_o)$  satisfies*

$$1. R_{u,v}^{\text{eff}}(\mathbf{Z}_1^{(h)}) = \Theta(d_{\mathbf{Z}_1}(u, v))$$

$$2. R_{u,v}^{\text{eff}}(\mathbf{Z}_2^{(h)}) = \Theta(\log d_{\mathbf{Z}_2}(u, v)),$$

$$3. R_{u,v}^{\text{eff}}(\mathbf{Z}_3^{(h)}) = \Theta(1).$$

□

*Proof of Lemma 5.5.2.* Consider the scalar electrical network  $(\mathbf{Z}_d, 1)$  formed by assigning a 1-Ohm resistance to every edge of the  $d$ -dimensional lattice  $\mathbf{Z}_d$ . The effective resistance between two nodes in the one-dimensional lattice network  $(\mathbf{Z}_1, 1)$  is given by  $r_{u,v}^{\text{eff}} = d_{\mathbf{Z}_1}(u, v)$ , which follows from series resistance formula. In the 2-dimensional lattice network  $(\mathbf{Z}_2, 1)$ , the effective resistances obeys  $r_{u,v}^{\text{eff}} = \Theta(\log d_{\mathbf{Z}_2}(u, v))$  [105]. Similarly, it was shown in [105] that for the scalar electrical network  $(\mathbf{Z}_3, 1)$ , the effective resistances obeys  $r_{u,v}^{\text{eff}} = \Theta(1)$ . The results now follow upon applying Lemma 4.6.1, which allows one to go from scalar effective resistances to matrix-valued case, and Lemma 5.5.1, which shows that the effective resistance in a graph and in its  $h$ -fuzz has the same order. ■

## 5.5.2 An intuitive explanation

Before proving the scaling laws for dense and sparse graphs, we offer an intuitive explanation, which comes from thinking of them as “coarse approximations” of the respective Euclidean spaces. In fact, thinking of the graph as a metric space, with the graphical distance being the associated metric, such approximations can be made rigorous if mappings between the node set and points on the Euclidean space are defined that preserve distances upto some constant factor (see [117] for a thorough exposition of this topic). A dense drawing of a graph in  $\mathbb{R}^d$  is essentially such a map, which ensures that the distortion (measured by the metric in the respective spaces, Euclidean or graphical) is upper-bounded

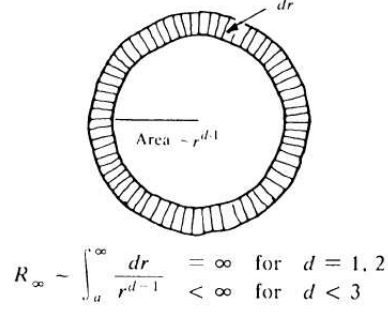


Figure 41: ♣

Figure 5.4. Doyle and Snell [2]’s illustration on approximating lattices by Euclidean spaces.

in going from the Euclidean metric space to the graphical metric space. This follows from Lemma 5.3.1.

In fact, the scaling laws can be explained by such coarse approximations, by examining the effective resistance in resistive medium filling the entire Euclidean space. The following quote from Doyle and Snell [2] explains it all:

*Suppose we replace our  $d$ -dimensional resistor lattice by a (homogeneous, isotropic) resistive medium filling all of  $\mathbb{R}^d$  and ask for the effective resistance to infinity. Naturally we expect that the rotational symmetry will make this continuous problem easier to solve than the original discrete problem. If we took this problem to a physicist, he or she would probably produce something like the scribbles illustrated in Figure 5.4<sup>2</sup>, and conclude that the effective resistance is infinite for  $d = 1, 2$  and finite for  $d > 2$ .*

Although Doyle and Snell [2] were concerned chiefly about resistances growing to infinity or staying bounded, we can conclude much more from continuum approximations, once we recognize that the matrix-valued effective resistance behaves quite similarly to the scalar valued one. By elementary calculations, one

---

<sup>2</sup>The figure number has been changed here for obvious reasons.

can conclude that the scalar effective resistance in a metallic rod grows as the length of the rod. This is shown in Figure 5.4. In an annular plate with inner radius  $r_o$  and outer radius  $r$ , the effective resistance between the inner and outer boundaries of the plate is an logarithmic function of the radius  $r$ , when  $r$  is large. In a sphere, however, similar calculations show that the effective resistance stays bounded by a constant even the size of the sphere is increased without bound. It is not difficult to convince ourselves that the  $d$ -dimensional lattice is a good approximation of the  $d$ -dimensional Euclidean space; hence it is no surprise that the effective resistances grow in the lattice  $\mathbf{Z}_d$  at the same rate as they do in  $\mathbb{R}^d$ . The results on effective resistances in lattices have been established rigorously in [105, 118].

In going from lattices to more general measurement graphs, we used denseness and sparseness properties to compare them to lattices by using embedding. However, a better understanding is obtained by comparing dense and sparse graphs to the Euclidean spaces. A graph that is dense in  $\mathbb{R}^d$  is essentially an “upper-approximation” of the Euclidean space  $\mathbb{R}^d$ , in the sense that when the graph is looked at through blurring lenses, it looks at least as dense as  $\mathbb{R}^d$ . Since the graphs has “more conductive material” than  $\mathbb{R}^d$ , the current faces less resistance and hence the effective resistance in the graph grows slowly compared to the effective resistance in the Euclidean space. Figure 5.2 also attempts to argue this pictorially. Similarly, a graph that is sparse in  $\mathbb{R}^d$  is a “lower-approximation” of  $\mathbb{R}^d$  – it is at least as sparse as  $\mathbb{R}^d$ . Since the graph has “less conductive material” than  $\mathbb{R}^d$ , the effective resistance in the graphs grows at least as fast as in  $\mathbb{R}^d$ . Such approximation is not uncommon in other fields of study. The field of coarse geometry, for example, assumes such a point of view and avers “...two spaces that look the same from a great distance are actually equivalent” [119].

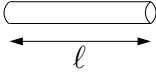
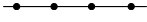
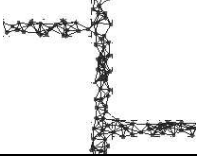
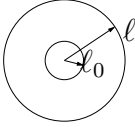
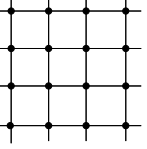
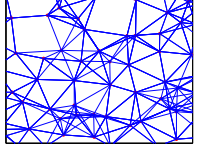

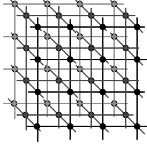
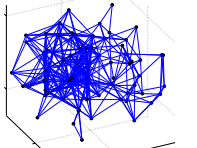
Continuous media and its $R^{\text{eff}}$		Lattice	Dense and sparse graphs
	$\int_{\ell_o}^{\ell} dr \sim \ell$		
	$\int_{\ell_o}^{\ell} \frac{dr}{2\pi r} \sim \log \ell$		
	$\int_{\ell_o}^{\ell} \frac{dr}{2\pi r^2} \sim 1$		

Table 5.2. An intuitive explanation of error scaling laws by continuum approximation. The notation  $f(x) \sim x$  means the ratio  $f(x)/x$  goes to a constant as  $x$  goes to infinity.

### 5.5.3 Proof of the Error Scaling Laws Theorem 5.3.1

We now prove Theorem 5.3.1 by using all the tools that have been developed in this chapter and in the previous one. The following terminology will be needed in the proofs. For a matrix-valued function  $g : \mathbb{R} \rightarrow \mathbb{R}^{k \times k}$  and a scalar-valued function  $p : \mathbb{R} \rightarrow \mathbb{R}$ , the notation  $g(y) = \Theta(p(y))$  means that  $g(y) = \Omega(p(y))$  and  $g(y) = \mathcal{O}(p(y))$ . The asymptotic notations  $\mathcal{O}$  and  $\Omega$  are described in Section 5.3.

*Proof of Theorem 5.3.1.* [Upper bounds:] Throughout the proof of the upper bounds, we will use  $R_{u,v}^{\text{eff}}(\mathcal{G})$ , for any graph  $\mathcal{G}$ , to denote the effective resistance between nodes  $u$  and  $v$  in the electrical network  $(\mathcal{G}, P_{\max})$  with every edge of  $\mathcal{G}$  having a generalized resistance of  $P_{\max}$ . Consider the generalized electrical network  $(\mathcal{G}, P_{\max})$  formed by assigning a constant generalized resistances of  $P_{\max}$  to

every edge of the measurement graph  $\mathcal{G}$ . From the Electrical Analogy theorem and Monotonicity Law (theorems 4.5.1 and 4.6.1), we get

$$\Sigma_{u,o} \leq R_{u,o}^{\text{eff}}(\mathcal{G}).$$

Since  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ , there exist a dense drawing of  $\mathcal{G}$  in  $\mathbb{R}^d$ , which we denote by a drawing function  $f$ , and a positive integer  $h$  such that the  $d$ -D lattice  $\mathbf{Z}_d$  can be embedded in the  $h$ -fuzz of  $\mathcal{G}$ . Moreover, Theorem 5.4.1 tells us that there exists  $u_z, o_z \in \mathcal{V}_{\mathbf{Z}_d}$ , a positive constant  $c$ , and an embedding  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$  of  $\mathbf{Z}_d$  into  $\mathcal{G}^{(h)}$ , such that

$$d_{\mathcal{G}}(u, \eta(u_z)) \leq c, \quad d_{\mathcal{G}}(o, \eta(o_z)) \leq c \quad (5.3)$$

$$d_{\mathbf{Z}_d}(u_z, o_z) < 4d + \frac{\sqrt{d}}{\gamma} d_f(u, o), \quad (5.4)$$

where  $\gamma$  is the maximum uncovered diameter of the  $f$ -drawing of  $\mathcal{G}$ . Note that  $\eta(u_z), \eta(o_z) \in \mathcal{V}$ . Consider the electrical network  $(\mathcal{G}^{(h)}, P_{\max})$  formed by assigning every edge of  $\mathcal{G}^{(h)}$  a resistance of  $P_{\max}$ . From the Triangle Inequality for effective resistances (Lemma 4.6.2),

$$\begin{aligned} R_{u,o}^{\text{eff}}(\mathcal{G}^{(h)}) &\leq R_{u, \eta(u_z)}^{\text{eff}}(\mathcal{G}^{(h)}) + R_{\eta(u_z), \eta(o_z)}^{\text{eff}}(\mathcal{G}^{(h)}) \\ &\quad + R_{\eta(o_z), o}^{\text{eff}}(\mathcal{G}^{(h)}). \end{aligned} \quad (5.5)$$

For any two nodes  $u, v \in \mathcal{V}$ , triangle inequality gives us  $R_{u,v}^{\text{eff}}(\mathcal{G}^{(h)}) \leq d_{\mathcal{G}^{(h)}}(u, v) P_{\max} \leq \frac{2}{h} d_{\mathcal{G}}(u, v) P_{\max}$ . Using this bound in (5.5), and by using (5.3), we get

$$R_{u,o}^{\text{eff}}(\mathcal{G}^{(h)}) \leq \frac{4c}{h} P_{\max} + R_{\eta(u_z), \eta(o_z)}^{\text{eff}}(\mathcal{G}^{(h)}). \quad (5.6)$$

Since  $\mathcal{G}^{(h)} \supset \mathbf{Z}_d$ , from Rayleigh's Monotonicity Law (Theorem 4.6.1), we get

$$R_{\eta(u_z), \eta(o_z)}^{\text{eff}}(\mathcal{G}^{(h)}) \leq R_{u_z, o_z}^{\text{eff}}(\mathbf{Z}_d).$$

When  $\mathcal{G}$  is dense in, say, 2D, we have from Lemma 5.5.2 that

$$R_{u_z, o_z}^{\text{eff}}(\mathbf{Z}_2) = \Theta(\log d_{\mathbf{Z}_2}(u_z, o_z)),$$

which implies

$$R_{\eta(u_z), \eta(o_z)}^{\text{eff}}(\mathcal{G}^{(h)}) = \mathcal{O}(\log d_{\mathbf{Z}_2}(u_z, o_z)).$$

Combining this with (5.4) and (5.6), we get

$$R_{u, o}^{\text{eff}}(\mathcal{G}^{(h)}) = \mathcal{O}(\log d_f(u, o)).$$

Since  $\mathcal{G}$  is a bounded degree graph, from Lemma 5.5.1 we know that the effective resistance in  $\mathcal{G}$  and its  $h$ -fuzz is of the same order:

$$R_{u, o}^{\text{eff}}(\mathcal{G}) = \Theta(R_{u, o}^{\text{eff}}(\mathcal{G}^{(h)})),$$

which gives us the desired result that, when  $\mathcal{G}$  is dense in 2D,

$$\Sigma_{u, o} \leq R_{u, o}^{\text{eff}}(\mathcal{G}) = \mathcal{O}(\log d_f(u, o)).$$

The statements of the upper bounds for 1 and 3-dimensions can be proved similarly. This concludes the proof of the upper bounds in Theorem 5.3.1.

[Lower bounds:] Throughout the proof of the lower bounds, for any graph  $\mathcal{G}$ , we will use  $R_{u, v}^{\text{eff}}(\mathcal{G})$  to denote the effective resistance between nodes  $u$  and  $v$  in the electrical network  $(\mathcal{G}, P_{\min})$  with every edge of  $\mathcal{G}$  having a generalized resistance of  $P_{\min}$ . Now consider the generalized electrical network  $(\mathcal{G}, P_{\min})$  where  $\mathcal{G}$  is the measurement graph  $\mathcal{G}$ . From the Electrical Analogy and Rayleigh's Monotonicity Law (theorems 4.5.1 and 4.6.1), we get

$$\Sigma_{u, o} \geq R_{u, o}^{\text{eff}}(\mathcal{G}). \tag{5.7}$$

Since  $\mathcal{G}$  is sparse in  $\mathbb{R}^d$ , it follows from Theorem 5.4.2 that there exists a positive integer  $h$ , such that  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$ . Let  $\eta : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}^{(h)}$  be the embedding of  $\mathcal{G}$  into

$\mathbf{Z}_d^{(h)}$ . Consider the generalized electrical network  $(\mathbf{Z}_d^{(h)}, P_{\min})$  formed by assigning a generalized resistance of  $P_{\min}$  to every edge of  $\mathbf{Z}_d^{(h)}$ . From Rayleigh's monotonicity law, we get

$$R_{u,o}^{\text{eff}}(\mathcal{G}) \geq R_{u_z,o_z}^{\text{eff}}(\mathbf{Z}_d^{(h)}), \quad (5.8)$$

where  $u_z = \eta(u), o_z = \eta(o)$  refer to the nodes in  $\mathbf{Z}_d^{(h)}$  that correspond to the nodes  $u, o$  in  $\mathcal{G}$ . When the graph is sparse in, say, 2D, it follows from (5.8) and Lemma 5.5.2 that

$$\begin{aligned} R_{u,o}^{\text{eff}}(\mathcal{G}) &= \Omega(\log d_{\mathbf{Z}_2}(u_z, o_z)) \\ &= \Omega(\log d_f(u, o)), \end{aligned}$$

where the second statement follows from (5.2) in Theorem 5.4.2. Combining the above with (5.7), we get  $\Sigma_{u,o} = \Omega(\log d_f(u, o))$ , which proves the lower bound in the 2D case. The statements for the lower bounds in the 1D and 3D can be proved in an analogous manner. This concludes the proof of the theorem.  $\blacksquare$

## 5.6 Comments and open problems

We established a classification of graphs, namely, dense or sparse in  $\mathbb{R}^d$ ,  $1 \leq d \leq 3$ , that determines how the optimal estimator error of a node grows with its distance from the reference node. The classification of dense and sparse graphs is interesting only for infinite graphs, since no finite graph can be dense in any dimension and a finite graph is sparse in every dimension.

Although infinite graphs are a compelling approximation to large finite graphs, this approximation puts a constraint that the scaling laws apply only to nodes in the interior of the finite graphs. For a node that is not close to the boundary

of the graph, the graph appears as if it extends to infinity in all directions. In that case we can regard the node as belonging to an infinite graph. In fact, the results in the Chapter 4 show that in such a situation, the BLUE covariance of the node in the finite measurement graph can be quite close to the covariance in the infinite graph. Still, it leaves the question of covariances of nodes in the boundary open. In a finite graph, it may be of interest to obtain bounds on the *maximum* BLUE covariance. This is equivalent to obtaining bounds on the maximum effective resistance in the graph, where the maximum is taken over all pairs of nodes. It may be possible to obtain such bounds for very special classes of finite graphs such as rectangular grids by using known results on the effective resistance in such graphs [120]. However, how to do it for a wider class of finite graphs is an open question.

Another avenue of future research is to examine the role of randomness in the graph's structure explicitly. Although the dense and sparse classification we obtained does allow randomness in the structure of the graph, the effect of such randomness on the scaling laws for the error is not explicitly accounted for in the present work. A useful research direction would be the investigation of the estimation error covariances in graphs with random structure, such as random geometric graphs [109]. The BLUE covariance itself is a random variable in these situations. Another interesting avenue is the exploration of BLUE covariances in scale-free graphs, which have been a popular – if somewhat controversial – model for many large scale networks, both man-made and natural [108].

Scale-free networks have the additional difficulty that there is no well-accepted definition of a scale free graph (see [121] for a extensive discussion on this issue). A preliminary investigation of a related class of graphs called Gromov hyperbolic graphs, which have the advantage of at least being well-defined, have been un-

dertaken in [122]. Effective resistance in small world graphs have been studied in [123] using methods of statistical physics.

## 5.7 Proofs

*Proof of Lemma 5.3.1.* We prove that 1 implies 2 by contradiction. Assuming that 2 does not hold, we have that

$$\forall \alpha \quad \forall \beta \quad \exists \bar{u}, \bar{v} \in \mathcal{V} \text{ such that } d_{\mathcal{G}}(\bar{u}, \bar{v}) > \alpha d_f(\bar{u}, \bar{v}) + \beta.$$

or equivalently

$$\forall \alpha \quad \forall \beta \quad \exists \bar{u}, \bar{v} \in \mathcal{V}$$

such that

$$\frac{d_f(\bar{u}, \bar{v})}{d_{\mathcal{G}}(\bar{u}, \bar{v})} < \frac{1}{\alpha} - \frac{\beta}{\alpha d_{\mathcal{G}}(\bar{u}, \bar{v})}.$$

This means that for a given  $\alpha, \beta$ , the set

$$\left\{ \frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} : u, v \in \mathcal{V} \text{ and } d_{\mathcal{G}}(u, v) \geq \beta \right\}$$

contains at least the element

$$\frac{d_f(\bar{u}, \bar{v})}{d_{\mathcal{G}}(\bar{u}, \bar{v})} < \frac{1}{\alpha} - \frac{\beta}{\alpha d_{\mathcal{G}}(\bar{u}, \bar{v})} < \frac{1}{\alpha}$$

and therefore

$$\inf \left\{ \frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} : u, v \in \mathcal{V} \text{ and } d_{\mathcal{G}}(u, v) \geq \beta \right\} < \frac{1}{\alpha}.$$

Making  $\beta \rightarrow \infty$  we obtain that

$$\rho = \lim_{\beta \rightarrow \infty} \inf \left\{ \frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} : u, v \in \mathcal{V} \text{ and } d_{\mathcal{G}}(u, v) \geq \beta \right\} < \frac{1}{\alpha}.$$

But since  $\alpha$  can be arbitrarily large, the above actually implies that  $\rho = 0$ , which contradicts 1.

To prove that 2 implies 1, we note that when 2 holds, we conclude that for every pair of nodes  $u, v \in \mathcal{V}$ , for which  $d_{\mathcal{G}}(u, v) \geq n$ , we have that

$$\frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} \geq \frac{1}{\alpha} - \frac{\beta}{d_{\mathcal{G}}(u, v)} \geq \frac{1}{\alpha} - \frac{\beta}{n}, \quad \forall u \neq v \in \mathcal{V}.$$

Therefore,

$$\inf \left\{ \frac{d_f(u, v)}{d_{\mathcal{G}}(u, v)} : u, v \in \mathcal{V} \text{ and } d_{\mathcal{G}}(u, v) \geq n \right\} \geq \frac{1}{\alpha} - \frac{\beta}{n}.$$

As  $n \rightarrow \infty$ , the left-hand side converges to  $\rho$  and the right-hand side converges to  $\frac{1}{\alpha} > 0$ , from which 1 follows.

The statements about the maximum connected range and the existence of constants  $\alpha > 0$ ,  $\beta \geq 0$  can be proved in a manner similar to that above. ■

*Proof of Theorem 5.4.1.* We will denote by  $g : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathbb{R}^d$  the natural drawing of the lattice  $\mathbf{Z}_d$ .

( $\Rightarrow$ ) We have to prove that if  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ , conditions (i) and (ii) are satisfied. Since  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ , there is a drawing function  $f : \mathcal{V} \rightarrow \mathbb{R}^d$  so that the  $f$ -drawing of  $\mathcal{G}$  has a  $\gamma < \infty$  and  $\rho > 0$ . Define a new drawing  $f' : \mathcal{V} \rightarrow \mathbb{R}^d$  as

$$f'(u) = \frac{1}{\gamma} f(u), \quad \forall u \in \mathcal{V},$$

so that the maximum uncovered diameter  $\gamma'$  of the  $f'$  drawing of  $\mathcal{G}$  is 1. Note that  $f'$  is still a dense drawing of  $\mathcal{G}$ . Now we superimpose the natural  $g$ -drawing of  $\mathbf{Z}_d$  on the  $f'$ -drawing of  $\mathcal{G}$ , and draw open balls of diameter one  $B(g(u_z), \frac{1}{2})$  centered at the natural drawing  $g(u_z)$  of every lattice node. Figure 5.5 shows an example in  $\mathbb{R}^2$ . Since  $\gamma' = 1$ , it follows from the definition of denseness that

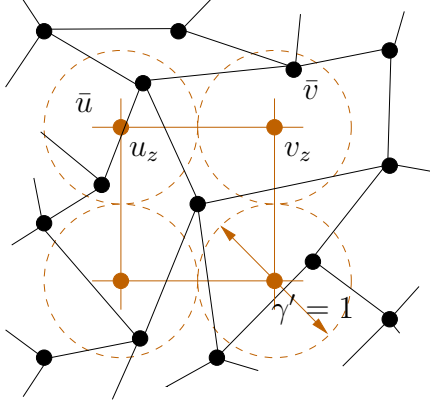


Figure 5.5. Superimposing a 2-dimensional lattice (brown) on a 2-dimensional dense graph (black).

in every one of those balls, there is at least one node  $u \in \mathcal{V}$ . To construct the embedding, we associate each node of the lattice to a node of  $\mathcal{G}$  whose drawing appears inside the ball centered around the lattice node. This defines an injective function  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$ . Consider two nodes of the lattice  $u_z, v_z \in \mathcal{V}_{\mathbf{Z}_d}$  that have an edge between them. Let  $\bar{u} := \eta(u_z), \bar{v} := \eta(v_z)$ . Since  $f'(\bar{u})$  and  $f'(\bar{v})$  belong to adjacent balls of unit diameter (see Figure 5.5),

$$d_{f'}(\bar{u}, \bar{v}) = \|f'(\bar{u}) - f'(\bar{v})\| \leq 2.$$

From Lemma 5.3.1 and  $f'$  being a dense drawing in  $\mathbb{R}^d$ , it follows that  $d_{\mathcal{G}}(\bar{u}, \bar{v}) \leq 2\alpha + \beta$ , for some positive constants  $\alpha$  and  $\beta$ . Define  $h := \lceil 2\alpha + \beta \rceil$ . Then  $\bar{u}$  and  $\bar{v}$  will have an edge between them in the  $h$ -fuzz  $\mathcal{G}^{(h)}$ . So  $\mathcal{G}^{(h)} \supset \mathbf{Z}_d$ , and we have the desired result that denseness implies (i).

For every  $u \in \mathcal{V}$ , find  $u_z \in \mathcal{V}_{\mathbf{Z}_d}$  as the node in the lattice such that the ball of unit diameter drawn around it is closest to  $u$ . That is, find  $u_z \in \mathcal{V}_{\mathbf{Z}_d}$  such that

$$u_z = \arg \min_{u'_z \in \mathcal{V}_{\mathbf{Z}_d}} \text{dist} \left( f'(u), B(g(u'_z), 1/2) \right) \quad (5.9)$$

where  $\text{dist}(x, A)$  between a point  $x \in \mathbb{R}^d$  and a set  $A \subset \mathbb{R}^d$  is defined as

$$\text{dist}(x, A) = \inf_{y \in A} \|x - y\|.$$

There are only  $2^d$  balls one needs to check to determine the minimum in (5.9), so  $u_z$  exists, though it may not be unique. If there are multiple minima, pick any one. This procedure defines an onto map  $\xi : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$ . Let  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$  be the embedding of  $\mathbf{Z}_d$  into  $\mathcal{G}^{(h)}$  as described earlier in this proof. Define  $\psi : \mathcal{V} \rightarrow \mathcal{V}$  as  $\psi := (\eta \circ \xi)$ . We will now show that, for every  $u \in \mathcal{V}$ , the node  $\psi(u) \in \mathcal{V}$ , which has a corresponding node in the lattice, is within a uniformly bounded graphical distance of  $u$ . Since  $f'(u)$  either lies in the ball centered at  $g(u_z)$  or in the gaps between that ball and the neighboring balls,  $\|f'(u) - g(u_z)\| < \sqrt{d}$ . Therefore,

$$\begin{aligned} d_{f'}(u, \psi(u)) &\leq \|f'(u) - g(u_z)\| + \|g(u_z) - f'(\psi(u))\| \\ &< \sqrt{d} + \frac{1}{2} \leq \frac{3}{2}\sqrt{d}, \end{aligned} \tag{5.10}$$

where we have used the fact that  $f'(\psi(u)) \in B(g(u_z), \frac{1}{2})$ . From Lemma 5.3.1 and the denseness of the  $f$ -drawing of  $\mathcal{G}$ , we get

$$\begin{aligned} d_{\mathcal{G}}(u, \psi(u)) &\leq \alpha d_f(u, \psi(u)) + \beta \\ &= \alpha \gamma d_{f'}(u, \psi(u)) + \beta \\ &< \frac{3}{2} \alpha \gamma \sqrt{d} + \beta. \end{aligned}$$

Define

$$c := \lceil \frac{3}{2} \alpha \gamma \sqrt{d} + \beta \rceil, \tag{5.11}$$

which is a constant independent of  $u$  and  $v$ . Then for every  $u \in \mathcal{V}$ , there exists a  $\bar{u} := \psi(u) \in \eta(\mathcal{V}_{\mathbf{Z}_d}) \subset \mathcal{V}$  such that  $d_{\mathcal{G}}(u, \bar{u}) < c$ , which is the desired condition (ii).

( $\Leftarrow$ ) We have to prove that if (i) and (ii) are satisfied, then  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ . We will construct a drawing  $f$  of  $\mathcal{G}$  in  $\mathbb{R}^d$  with the following procedure and then prove that it is a dense drawing. Since  $\mathbf{Z} \subset \mathcal{G}^{(h)}$ , there is an injective map  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$  such that  $\eta(\mathcal{V}_{\mathbf{Z}_d}) \subset \mathcal{V}$ . Pick a node  $u$  in  $\mathcal{V}$  that has not been drawn yet. By (ii), there exists a positive constant  $c$  and a node  $u_z \in \mathcal{V}_{\mathbf{Z}_d}$  such that  $\bar{u} := \eta(u_z) \in \mathcal{V}$  and  $d_{\mathcal{G}}(u, \bar{u}) < c$ . If  $\bar{u}$  has not been drawn yet, then draw it the location of its corresponding lattice node, i.e.,

$$f(\bar{u}) = g(u_z). \quad (5.12)$$

A little thought will reveal that if  $\bar{u}$  has been drawn already, as long as the drawing procedure outlined so far is followed, it must have been drawn on the lattice location  $g(u_z)$ , so (5.12) holds. Once  $\bar{u}$  is drawn, we draw  $u$  in the following way. In case  $\bar{u} = u$ , drawing of  $u$  is determined by the drawing of  $\bar{u}$ . If  $u \neq \bar{u}$ , draw  $u$  by choosing a random location inside an open ball of diameter 1 with the center at  $f(\bar{u})$ . To show that a drawing obtained this way is dense, first note that the largest uncovered diameter  $\gamma < 2$  since a subset of the nodes of  $\mathcal{V}$  occupy the lattice node positions. Pick any two nodes  $u, v \in \mathcal{V}$ . Again, from (ii), we know that there exists  $\bar{u}, \bar{v} \in \eta(\mathcal{V}_{\mathbf{Z}_d}) \subset \mathcal{V}$  such that  $d_{\mathcal{G}}(u, \bar{u}) \leq c$  and  $d_{\mathcal{G}}(v, \bar{v}) \leq c$  for some positive constant  $c$ . Therefore

$$\begin{aligned} d_{\mathcal{G}}(u, v) &\leq d_{\mathcal{G}}(u, \bar{u}) + d_{\mathcal{G}}(\bar{u}, \bar{v}) + d_{\mathcal{G}}(\bar{v}, v) \\ &\leq 2c + h d_{\mathcal{G}^{(h)}}(\bar{u}, \bar{v}) \end{aligned}$$

Since  $\mathbf{Z}_d \subset \mathcal{G}^{(h)}$ ,

$$\begin{aligned} d_{\mathcal{G}^{(h)}}(\bar{u}, \bar{v}) &\leq d_{\mathbf{Z}_d}(\eta^{-1}(\bar{u}), \eta^{-1}(\bar{v})) \\ &= \|g(u_z) - g(v_z)\|_1 \end{aligned}$$

where  $\|\cdot\|_1$  denotes the vector 1-norm,

$$\begin{aligned}
&\leq \sqrt{d}\|g(u_z) - g(v_z)\| \\
&= \sqrt{d}\|f(\bar{u}) - f(\bar{v})\| \text{ (from (5.12))} \\
&= \sqrt{d}d_f(\bar{u}, \bar{v}).
\end{aligned}$$

Because of the way the drawing  $f$  is constructed, we have  $d_f(u, \bar{u}) \leq 1$ , which implies  $d_f(\bar{u}, \bar{u}) \leq d_f(\bar{u}, u) + d_f(u, v) + d_f(v, \bar{v}) = d_f(u, v) + 2$ . So we have

$$\begin{aligned}
d_{\mathcal{G}}(u, v) &\leq 2c + h\sqrt{d}(d_f(u, v) + 2) \\
&= 2(c + h\sqrt{d}) + h\sqrt{d}d_f(u, v).
\end{aligned}$$

From Lemma 5.3.1, we see that the asymptotic distance ratio  $\rho > 0$  for the  $f$ -drawing of  $\mathcal{G}$ , which establishes that  $f$  is a dense drawing of  $\mathcal{G}$  in  $\mathbb{R}^d$ . It follows that  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ .

To prove the relationship (5.1) for any dense drawing  $f$ , consider again the scaled drawing  $f'$  defined as  $f' = f/\gamma$ , so that the maximum uncovered diameter of  $f'$  is 1. Since  $\mathcal{G}$  is dense in  $\mathbb{R}^d$ ,  $\mathbf{Z}_d$  can be embedded in  $\mathcal{G}^{(h)}$  with an embedding  $\eta : \mathcal{V}_{\mathbf{Z}_d} \rightarrow \mathcal{V}$ . We choose the embedding  $\eta$  as described in the first part of the proof. For every  $u \in \mathcal{V}$ , call  $u_z := \xi(u)$ , where  $\xi : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$  was defined earlier in this proof for the  $f'$  dense drawing of  $\mathcal{G}$ . Now consider two arbitrary nodes  $u, v \in \mathcal{V}$  and let  $u_z := \xi(u)$ ,  $v_z := \xi(v)$  (see Figure 5.6). It was shown earlier in this proof that for every pair of nodes  $u, v \in \mathcal{V}$ , we have  $d_{\mathcal{G}}(u, \eta(u_z)) < c$  and  $d_{\mathcal{G}}(v, \eta(v_z)) < c$ , where  $c$  is defined in (5.11).

Now,

$$\begin{aligned}
d_{\mathbf{Z}_d}(u_z, v_z) &= \|g(u_z) - g(v_z)\|_1 \\
&\leq \sqrt{d}\|g(u_z) - g(v_z)\|,
\end{aligned}$$

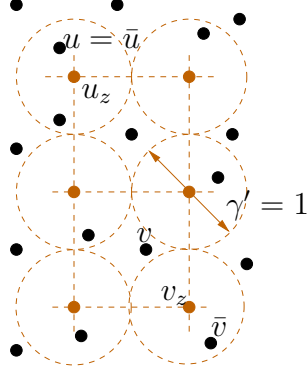


Figure 5.6. Natural drawing of the 2-D lattice (brown) superimposed on the  $f'$  drawing of  $\mathcal{G}$ . Edges are not shown to prevent clutter. In this example,  $u = \bar{u}$  but  $v \neq \bar{v}$ .

and

$$\begin{aligned} \|g(u_z) - g(v_z)\| &\leq \|g(u_z) - f'(\bar{u})\| + \|f'(\bar{u}) - f'(u)\| + \\ &\quad \|f'(u) - f'(v)\| + \|f'(v) - f'(\bar{v})\| + \\ &\quad \|f'(\bar{v}) - g(v_z)\|. \end{aligned}$$

We know that  $\|g(u_z) - f'(\bar{u})\| \leq \frac{1}{2} \leq \frac{\sqrt{d}}{2}$  since  $f'(\bar{u}) \in B(g(u_z), \frac{1}{2})$ , and  $\|f'(u) - f'(\bar{u})\| < \frac{3}{2}\sqrt{d}$  from (5.10). Using these in the above, we get

$$\begin{aligned} \|g(u_z) - g(v_z)\| &\leq 4\sqrt{d} + d_{f'}(u, v), \\ \Rightarrow d_{\mathbf{Z}_d}(u_z, v_z) &\leq 4d + \frac{\sqrt{d}}{\gamma} d_f(u, v) \end{aligned}$$

which is the desired result. ■

*Proof of Theorem 5.4.2.* We will denote by  $g : \mathcal{V}'_{\mathbf{Z}_d} \rightarrow \mathbb{R}^d$  the natural drawing of the lattice  $\mathbf{Z}_d$ .

$\Rightarrow$  Since  $\mathcal{G}$  is sparse in  $\mathbb{R}^d$ , there is a drawing function  $f : \mathcal{V} \rightarrow \mathbb{R}^d$  that produces a civilized drawing of  $\mathcal{G}$  with minimum node distance  $s > 0$  and maximum connected range  $r < \infty$ . Consider a new drawing  $f' : \mathcal{V} \rightarrow \mathbb{R}^d$  of  $\mathcal{G}$  defined as

$$f'(u) = \frac{\sqrt{d}}{s} f(u), \forall u \in \mathcal{V}. \quad (5.13)$$

The minimum node distance and the maximum connected range in this drawing are

$$s' = \sqrt{d}, \quad r' = \frac{\sqrt{d}}{s} r.$$

Superimpose the two drawings  $g(\mathbf{Z}_d)$  and  $f'(\mathcal{G})$  (cf. figure 5.7). In every lattice cell<sup>3</sup> in  $\mathbb{R}^d$ , there is at most one node of  $\mathcal{G}$ , for if there are two, then in an open ball of diameter  $\sqrt{d}$  in  $\mathbb{R}^d$ , there are two points  $f'(u)$  and  $f'(v)$  where  $u, v \in \mathcal{V}$ , which violates the condition that

$$s' = \inf_{\substack{u \neq v \\ u, v \in \mathcal{V}}} \|f'(u) - f'(v)\| = \sqrt{d}.$$

Define a mapping  $\eta : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$  by associating every node  $u \in \mathcal{V}$  to the lattice vertex with the most negative coordinate in the lattice cell where  $f'(u)$  lies (cf. Figure 5.7). Since a lattice cell contains the  $f'$  drawing at most one node,  $\eta$  is injective. Let  $u_z := \eta(u)$  and  $v_z := \eta(v)$ . Since  $f'(u)$  and  $g(u_z)$  lie in the same lattice cell,

$$\|f'(u) - g(u_z)\| \leq \sqrt{d}, \quad \forall u \in \mathcal{V}. \quad (5.14)$$

---

<sup>3</sup>A lattice cell is taken as a unit semi open hypercube in  $\mathbb{R}^d$ , which is a subset of  $\mathbb{R}^d$  of the form  $[a_1, a_1 + 1) \times [a_2, a_2 + 1) \cdots \times [a_d, a_d + 1)$  for some  $a_1, a_2, \dots, a_d \in \mathbb{R}$ .

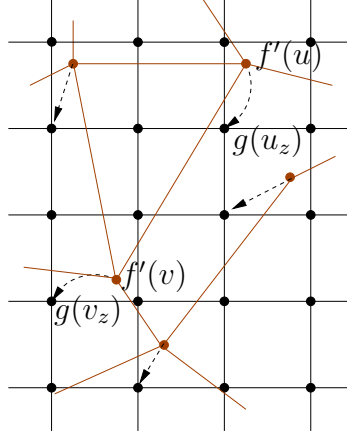


Figure 5.7. A graph that can be drawn in a civilized manner in  $\mathbb{R}^2$  can be embedded in a fuzz of the 2-D lattice.

So,

$$\begin{aligned}
 d_{\mathbf{Z}_d}(u_z, v_z) &= \|g(u_z) - g(v_z)\|_1 \\
 &\leq \sqrt{d} \|g(u_z) - g(v_z)\| \\
 &\leq \sqrt{d} \left( \|g(u_z) - f'(u)\| + \|f'(u) - f'(v)\| \right. \\
 &\quad \left. + \|f'(v) - g(v_z)\| \right) \\
 &\leq \sqrt{d} \left( 2\sqrt{d} + d_{f'}(u, v) \right)
 \end{aligned}$$

If  $(u, v) \in \mathcal{E}$ ,  $d_{f'}(u, v) \leq r' = \frac{\sqrt{d}}{s}r$ , where  $r'$  is the maximum connected range in the  $f'$  drawing of  $\mathcal{G}$ , so we get

$$d_{\mathbf{Z}_d}(u_z, v_z) \leq 2d + \frac{d}{s}r < \infty$$

Defining  $h := \lceil 2d + \frac{d}{s}r \rceil$ , we see that there is an edge between  $u_z$  and  $v_z$ , i.e., between  $\eta(u)$  and  $\eta(v)$ , in  $\mathbf{Z}_d^{(h)}$ . This proves that  $\eta$  is an embedding, so  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$ .

$\Leftarrow$  Since  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$  for some  $h < \infty$ , there is an embedding  $\eta : \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$  of  $\mathcal{G}$  in  $\mathbf{Z}_d$ . Consider a drawing  $f$  of  $\mathcal{G}$  defined as

$$f(u) = g(\eta(u)), \quad \forall u \in \mathcal{V}.$$

We immediately get  $s \geq 1 > 0$  for this drawing. If  $(u, v) \in \mathcal{E}$ ,

$$\begin{aligned} \|f(u) - f(v)\| &= \|g(\eta(u)) - g(\eta(v))\| \\ &\leq \|g(\eta(u)) - g(\eta(v))\|_1 \\ &= d_{\mathbf{Z}_d}(\eta(u), \eta(v)) \leq h, \end{aligned}$$

where the last inequality follows because  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$ . Therefore the maximum connected range  $r$  in the drawing  $f$  of  $\mathcal{G}$  satisfies  $r \leq h < \infty$ . The drawing of  $\mathcal{G}$  specified by  $f$  is therefore a civilized drawing in  $\mathbb{R}^d$ , from which it follows that  $\mathcal{G}$  is sparse in  $\mathbb{R}^d$ .

To prove the relation (5.2), we go back to the drawing  $f'$  defined in (5.13) based on the civilized drawing  $f$  of  $\mathcal{G}$ . Since  $\mathcal{G}$  is sparse in  $\mathbb{R}^d$ ,  $\mathcal{G} \subset \mathbf{Z}_d^{(h)}$  for some positive integer  $h$ . Let  $\eta: \mathcal{V} \rightarrow \mathcal{V}_{\mathbf{Z}_d}$  be the embedding of  $\mathcal{G}$  into  $\mathbf{Z}_d^{(h)}$ . Denoting  $u_z := \eta(u)$ , for any  $u, v \in \mathcal{V}$  we get

$$\begin{aligned} d_{\mathbf{Z}_d}(u_z, v_z) &= \|g(u_z) - g(v_z)\|_1 \\ &\geq \|g(u_z) - g(v_z)\|. \end{aligned}$$

Since  $\|f'(u) - f'(v)\| \leq \|f'(u) - g(u_z)\| + \|g(u_z) - g(v_z)\| + \|g(v_z) - f'(v)\|$ , we get from the above that

$$\begin{aligned} d_{\mathbf{Z}_d}(u_z, v_z) &\geq \|f'(u) - f'(v)\| - \|f'(u) - g(u_z)\| - \|g(v_z) - f'(v)\| \\ &\geq d_{f'}(u, v) - 2\sqrt{d}. \end{aligned}$$

where we have used that fact that both  $f'(u)$  and  $g(u_z)$  lie in the same lattice cell for every  $u \in \mathcal{V}$ . Since  $d_{f'}(\cdot) = d_f(\cdot) \frac{\sqrt{d}}{s}$ , the result follows. ■

*Proof of Lemma 5.4.1.* We only provide the proof that the 2-dimensional lattice is not sparse in  $\mathbb{R}$  and is not dense in  $\mathbb{R}^3$ . The general case for arbitrary dimensions is analogous.

To prove by contradiction the lack of denseness, assume that there exists a dense drawing  $f$  of  $\mathbf{Z}_2$  in  $\mathbb{R}^3$ , with associated  $\gamma < \infty$  and  $\rho > 0$ . Fix the origin of  $\mathbb{R}^3$  at  $f(u)$  for an arbitrary node  $u$  in the lattice  $\mathbf{Z}_2$ . For an arbitrary  $D > 0$ , the volume of the sphere in  $\mathbb{R}^3$  centered at the origin with diameter  $D$ , denoted by  $\mathcal{B}^3(0, D)$  is  $\Omega(D^3)$ . Therefore the number of nodes of  $\mathbf{Z}_2$  drawn inside  $\mathcal{B}^3(0, D)$  is  $\Omega((\frac{D}{\delta})^3) = \Omega(D^3)$ . It is straightforward to show that for any set of  $n$  distinct nodes in the lattice  $\mathbf{Z}_2$ , the maximum graphical distance between any two nodes in the set is  $\Omega(\sqrt{n})$ . Therefore the maximum graphical distance between the nodes in  $\mathcal{B}^3(0, D)$  is  $\Omega(D^{\frac{3}{2}})$ .

The maximum Euclidean distance between any two nodes drawn inside the sphere  $\mathcal{B}^3(0, D)$  under the  $f$ -drawing is at most  $D$ , and since  $f$  is a dense drawing, it follows from Lemma 5.3.1 that for every pair of nodes  $u, v$  in  $\mathbf{Z}_2$  such that  $f(u), f(v) \in \mathcal{B}^3(0, D)$ , we have  $d_G(u, v) \leq aD + b$ . Therefore, the maximum graphical distance between pairs of nodes whose drawing falls inside  $\mathcal{B}^3(0, D)$  is  $\mathcal{O}(D)$ , as well as  $\Omega(D^{\frac{3}{2}})$ , which is a contradiction for sufficiently large  $D$ . Hence no dense drawing of  $\mathbf{Z}_2$  in  $\mathbb{R}^3$  is possible.

To show  $\mathbf{Z}_2$  is not sparse in  $\mathbb{R}$ , assume that there exists a civilized drawing of  $\mathbf{Z}_2$  in  $\mathbb{R}$  with  $s > 0$  and  $r < \infty$ , where  $r$  and  $s$  are constants. Consider a subgraph  $\mathbf{Z}_{2(n)}$  of  $\mathbf{Z}_2$  that consists of all nodes within a Euclidean distance  $n$  from the origin. The total number of nodes in this finite subgraph is  $\Omega(n^2)$ . The length of the interval,  $L$ , in which the nodes of this subgraph are located in the sparse 1-d drawing of  $\mathbf{Z}_2$  is clearly  $L = \Omega(sn^2)$ . Since the maximum graphical distance between any two nodes in the subgraph  $\mathbf{Z}_{2(n)}$  is  $n$  by construction, the maximum connected range in the 1-d drawing must be at least  $r \geq \frac{L}{n} = \Omega(sn)$ . Since this must be true for every  $n$ ,  $r$  cannot be a finite constant. Thus, no civilized drawing of  $\mathbf{Z}_2$  in  $\mathbb{R}$  exists. ■

*Proof of Lemma 5.5.1.* Due to Lemma 4.6.1, we need to prove the result only for the case of scalar-valued unit resistors. Let  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  be a connected graph with an unit resistor on every edge. Let  $\mathcal{G}^{(h)} = (\mathcal{V}, \mathcal{E}^{(h)})$  be the  $h$ -fuzz of  $\mathcal{G}$ . We assign an unit resistance to every edge of  $\mathcal{E}^{(h)}$ . The edge set  $\mathcal{E}^{(h)}$  consists of two disjoint subsets  $\mathcal{E}$  and  $\mathcal{E}_h$ , where  $\mathcal{E}_h$  is the set of “new” edges in  $\mathcal{G}^{(h)}$  that were not there in  $\mathcal{G}$ . That is,  $\mathcal{E}^{(h)} = \mathcal{E} \cup \mathcal{E}_h$  and  $\mathcal{E} \cap \mathcal{E}_h = \emptyset$ . To every edge  $e \in \mathcal{E}_h$ , there corresponds a path  $\mathcal{P}_e$  of length  $\ell_e \leq h$  in  $\mathcal{G}$ . See figure 5.8(A) and (B) for example. Replace every edge  $e \in \mathcal{E}_h$  by a series of  $\ell_e$  edges, each with unit resistance, and call the resulting graph  $\overline{\mathcal{G}^{(h)}}$ . We introduce new nodes in doing so. To every one of these new nodes in  $\overline{\mathcal{G}^{(h)}}$ , there corresponds a node in  $\mathcal{G}$  (see figure 5.8(C)). by Rayleigh’s monotonicity law, the effective resistance has increased:  $R_{\bar{u}, \bar{v}}^{\text{eff}}(\overline{\mathcal{G}^{(h)}}) \geq R_{u, v}^{\text{eff}}(\mathcal{G}^{(h)})$ , where  $\bar{u}$  is a node in  $\overline{\mathcal{G}^{(h)}}$  that corresponds to  $u$  in  $\mathcal{G}^{(h)}$ . However, since we have increased the resistance of any edge by no more than a factor of  $h$ , the increase in effective resistance is no more than a factor of  $h$ :

$$R_{\bar{u}, \bar{v}}^{\text{eff}}(\overline{\mathcal{G}^{(h)}}) \leq h R_{u, v}^{\text{eff}}(\mathcal{G}^{(h)}). \quad (5.15)$$

Now for every edge  $e$  of  $\mathcal{E}_h$ , look at the corresponding series of resistors in  $\overline{\mathcal{E}^{(h)}}$ . Its endpoints lie in the original graph  $\mathcal{G}$ . Take its intermediate vertices and short them to vertices of  $\overline{\mathcal{G}^{(h)}}$  along the path  $\mathcal{P}_e$ . We do this for every edge  $e \in \mathcal{E}_h$  and call the resulting graph  $\mathcal{G}'$ . Again due to Rayleigh’s monotonicity law,

$$R_{u', v'}^{\text{eff}}(\mathcal{G}') \leq R_{\bar{u}, \bar{v}}^{\text{eff}}(\overline{\mathcal{G}^{(h)}}), \quad (5.16)$$

where  $u'$  denotes the node in  $\mathcal{G}'$  that corresponds to  $\bar{u}$  in  $\overline{\mathcal{G}^{(h)}}$ . The graph  $\mathcal{G}'$  differs from  $\mathcal{G}$  only in having extra parallel edges between its nodes (figure 5.8(D)). However, the number of edges in  $\mathcal{G}'$  that are parallel to an edge  $e$  in  $\mathcal{G}$  are no more than the number of paths in  $\mathcal{G}$  of length at most  $h$  that traverse the edge  $e$ . Since  $\mathcal{G}$  is a bounded degree graph, there is an upper bound to this number. Let  $\eta$  be

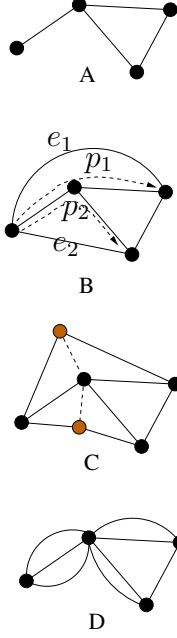


Figure 5.8. Fuzzing doesn't change the effective resistance too much.

this upper bound. It is easy to see from Parallel Resistors Proposition 4.6.1 that if every edge of a graph – that has unit resistance on every edge – is replaced by  $\eta$  parallel edges and every one of the new edges have unit resistance, then the effective resistance in the new graphs lower than that in the original graph by a factor of  $\eta$ . Combining with Rayleigh's monotonicity law, we get

$$\frac{1}{\eta} R_{u,v}^{\text{eff}}(\mathcal{G}) \leq R_{u',v'}^{\text{eff}}(\mathcal{G}'). \quad (5.17)$$

Equations (5.15), (5.16) and (5.17) give us

$$R_{u,v}^{\text{eff}}(\mathcal{G}^{(h)}) > \frac{1}{h\eta} R_{u,v}^{\text{eff}}(\mathcal{G}), \quad (5.18)$$

which proves the result. ■

## Part II

### Control with Relative Measurements

# Chapter 6

## Decentralized formation control: effective resistance vs. scalability

### 6.1 Introduction

In this chapter we consider the problem of formation control by a group of autonomous agents. A formation is specified in terms of a desired relative positions between agents. Each agent can measure its relative position with only a limited number of nearby agents. The task for each agent is to take control actions, e.g., modify its acceleration and/or velocity, using only the locally available information, such that the group attains its collective goal of maintaining the desired formation.

Motivation for studying formation control problems arise from its relevance to diverse applications, from military surveillance to swarming in nature. Keeping a formation among a group of vehicles is important in certain military applications where sensor assets are limited. In that case, individual team members

can concentrate their sensors across a limited portion of the environment, while the team as a whole is still able to sense the whole environment [23]. A group of aerial vehicles can reduce drag by maintaining their relative positions at specific values [18, 19]. Similarly, capacity of highways can be improved by if large groups of vehicles, called platoons, can move in formation maintaining a small inter-vehicular separation (see [6] and references therein). Yet another application of formation control is interferometric imaging by a formation of satellites that can lead to a higher degree of accuracy than what is possible by a single satellite [24, 124].

Study of formation control is useful in understanding biological systems as well. Several species of birds and many species of fish and aquatic animals are known to exhibit “swarming”, which loosely means some form of aggregate motion by a group as a whole. A few example of such swarming are: pattern forming by schools of fish [125], synchronized predation of cope-pods by juvenile herring [126], moving in formation by spiny lobsters to reduce drag [20] and V-formation flying by birds (apparently to improve lift [22] or for better visual cues [127]).

The control action taken by an agent necessarily depends on the states of the other agents in order to maintain the formation. Thus, the dynamics of individual agents become coupled, or interconnected, which can be described by a graph. The nodes of the graph are the agents and the edges are the node pairs whose states appear in each other’s control algorithms. We call this graph the *interconnection graph*.

The interconnection graph depends on the choice of control architecture. In a *centralized* control architecture, all the relative position measurements are made available to a either a leader node or to every node. If the control signals for all the nodes are computed by a leader, it transmits those signals back to the

individual nodes. Otherwise, every node computes its own control signal from the global information it has available. In the former case, the interconnection graph will look like a “star”, with an edge  $(u, o)$  between every node  $u$  and the leader. In the latter, the interconnection graph will be a complete graph, in which every node is connected to every other node.

In contrast, in a *decentralized* architecture, every node uses only the information that it can obtain with either communication with its nearby node or with on-board sensors. For example, when every node uses the measurements of its relative position with respect to its nearby nodes, which it can obtain using on-board sensors such as radars, the resulting architecture is decentralized. The interconnection graph will have a highly local structure, with edges existing only between nodes that are physically close. A centralized architecture suffers from a high communication overhead compared to a decentralized one, which makes decentralized architecture more appealing, particularly for large groups of autonomous agents.

We will study decentralized architectures in this chapter, and focus on situations when the interconnection graph has a large number of nodes. Scalability is an important issue, which refers to how sensitive the performance of the closed loop is to the number of nodes. The performance metric is application dependent, and may refer to the stability margin, sensitivity to measurement noise, etc. Typically, if the performance of the closed loop is independent of, or degrades slowly with, the number of nodes, then the control algorithm is termed scalable. We will see in this chapter that scalability of formation control is as much a function of the structure of the interconnection graph as it is of the control algorithm.

Since our focus is on the interconnection structure, we consider simple models of node dynamics and simple control laws. In particular, the dynamics of each

node is modeled as an integrator, and each node modifies its velocity depending on a local error it estimates by comparing the positions of its neighbors (relative to itself) to the desired ones. Such node models and control laws have been investigated extensively in the literature [29, 128]. We will refer to this particular control law as *Laplacian disagreement control*, the term being borrowed due to the control law's close connection to the *Laplacian disagreement function* used in [129]. Apart from employing a simple control law, we also ignore some of the issues faced in practice, such as sensing and communication faults, time variation in the interconnection due to these and other reasons, avoiding obstacles, etc.

*Chapter organization:* Two topics are studied in this chapter, that of sensitivity to measurement noise and the minimum eigenvalue of the Dirichlet Laplacian. Each section introduces a topic, states the problem and then presents the results. Section 6.3 investigates noise sensitivity of formation control with relative measurements. Section 6.4 describes a lower bound on the smallest eigenvalue of the Dirichlet Laplacian with effective resistances, along with a summary of applications where such bound is useful.

## 6.2 Contributions and prior work

The results established in this chapter are briefly summarized below:

1. Graph structure and error propagation: we show in Section 6.3 that the covariance of the steady-state formation error of a node is equal to the matrix-valued effective resistance between the node and the reference in an abstract electrical network constructed from the architecture graph. The formation error of a node is the difference between its relative position w.r.t. a reference

and the desired relative position. The matrix-valued effective resistance was introduced in Chapter 4. It was shown in Chapter 5 that graph structure has a huge impact on the effective resistance. This analogy with electrical networks show that the performance of the Laplacian disagreement control is quite sensitive to the structure of the interconnection graph.

2. Bound on the Dirichlet Laplacian spectrum: In Section 6.3 we derive a lower bound on the minimum eigenvalue of the Dirichlet Laplacian of a matrix-weighted graph in terms of the effective resistances. The Dirichlet Laplacian was introduced in Chapter 2. The stability margin of the closed loop formation depends on this eigenvalue of the interconnection graph. In addition, convergence rate of the Jacobi algorithm described in Chapter 3 also depends on this eigenvalue of the measurement graph. To determine the performance of these algorithms, we need a lower bound on this eigenvalue; whereas few results are available in the literature on lower bounds on the Dirichlet Laplacian spectrum. The bound derived here in terms of the effective resistances is useful when bounds on the effective resistance can be derived. This is possible for certain classes graphs without complete knowledge of the graph itself, when graph embedding techniques can be used to relate them to graphs with known effective resistance, as described in Chapters 4 and 5.

We see from the discussion above that the generalized effective resistance, introduced in Chapter 4 in connection with estimation problems, has a potential for fruitful use in control problems as well, especially in multi-agent coordination problems that are posed in terms of matrix-weighted graphs.

*Prior work:* Although the Laplacian disagreement control has been examined extensively in connection with consensus algorithms (see, e.g., [29, 30, 129]), the

effect of interconnection structure on its performance has received scant attention. Except in the special case of 1D formations of automated vehicular platoons, the effect of disturbances on spacing errors have not been thoroughly investigated. The platoon problem will be discussed extensively in the following two chapters.

Various forms of formation control using relative measurements have been examined, including behavior based approaches [23], control using artificial potentials, etc. Typical results in this area consists of a control algorithm and a proof of asymptotic stability. When vehicle dynamics or the control laws are non-linear [130–132], or when the interconnection graph is directed [133], establishing asymptotic stability itself is challenging. So examination of stability margins, especially for large groups of autonomous agents, yet to generate much enthusiasm. Although it has been recognized that the convergence rate of the errors will depend on the interconnection graph, most studies focus on “leaderless coordination”, in which the Laplacian eigenvalues that become important [128, 133]. This emphasis is partly due to the popularity of “consensus algorithms” and their close connection with formation control [30], in which leaderless coordination is natural. However, in contrast to consensus algorithms, in formation control, the presence of a reference node in the interconnection graph is common and realistic, since an average trajectory has to be specified to the whole formation in some form, which is usually done through either a lead vehicle or a virtual leader, such as in [134]. In that case, the Dirichlet Laplacian eigenvalue is the critical quantity.

The spectrum of the Dirichlet Laplacian has attracted very little attention from graph theorists. Except for a few results in [81], most work has focused on the Laplacian spectra. The minimum eigenvalue of the Dirichlet Laplacian is smaller than the second smallest eigenvalue of the graph Laplacian, the latter being also called the *algebraic connectivity*. Therefore, bounds on the algebraic connectivity,

on which extensive literature exists, are not useful in lower bounding the minimum eigenvalue of the Dirichlet Laplacian eigenvalues. The effective resistance based lower bound is therefore quite useful, especially for graphs for which the order estimates of the effective resistances are known.

### 6.3 Formation control with noisy measurements

Consider a group of  $N$  mobile nodes moving in  $k$ -dimensional space. One of the objectives of the group is to maintain a pre-specified formation defined by the relative positions between nodes. In particular, denoting by  $x_u \in \mathbb{R}^k$ ,  $u \in \mathcal{V} := \{1, 2, \dots, N\}$  the position of the  $u$ th node, the control objective is to make the positions converge to values for which

$$x_u - x_v = r_{u,v}, \quad \forall (u, v) \in \mathcal{V} \times \mathcal{V}, \quad (6.1)$$

where  $r_{u,v}$  denotes the desired relative position of node  $u$  with respect to node  $v$ .

Not all nodes are able to measure their relative positions with respect to all other nodes and therefore each node is constrained to use only a few relative position measurements to compute its control signal. We denote by  $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$  the set of ordered pairs of nodes that can measure their relative positions. In particular, the existence of a pair  $(u, v)$  in  $\mathcal{E}$  signifies that node  $u$  can measure its position with respect to  $v$ . Thus the group is formally described in terms of a directed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , whose nodes represents the nodes and whose edges represent the pairs of nodes that have access to a relative measurement. We assume that a noisy relative measurement  $y_{u,v}$  of the following form is available to node  $u$  if  $(u, v) \in \mathcal{E}$ :

$$y_{u,v}(t) = x_u(t) - x_v(t) + \epsilon_{u,v}(t) \quad (6.2)$$

where  $\epsilon_{u,v}$  is a white random noise process with auto-covariance

$$\mathbb{E}[\epsilon_{u,v}(t_1)\epsilon_{u,v}^T(t_2)] = \delta(t_1 - t_2)R_{u,v}. \quad (6.3)$$

One of the nodes  $o \in \mathcal{V}$  will be called the *reference* and it will move independently of the remaining ones. The remaining nodes attempt to maintain the formation specified by (6.1). The reference node may or may not be a physical agent. It may be a virtual reference that is known to at least one of the physical agents. In case  $x_o$  is not a physical agent, an edge between the node  $u$  and the reference  $o$  means that the agent  $u$  is able to measure its position with respect to the reference  $o$ . Or, the reference node maybe a proxy for the moving frame of reference if all the nodes are moving at a constant velocity.

We assume the following:

**Assumption 6.3.1.** 1. The graph  $\mathcal{G}$  is time-invariant.

2. If  $(u, v) \in \mathcal{E}$ , then  $(v, u) \in \mathcal{E}$ .
3. The noise processes over different edges are independent of each other, i.e.,  $e_{u,v}(t)$  is independent of  $e_{v,u}(t)$  for all  $t \in \mathbb{R}^+$ .
4. Even though the measurement errors on the two edges  $(u, v)$  and  $(v, u)$  connecting the nodes  $u$  and  $v$  are uncorrelated, they have the same auto-covariance matrix; i.e.,  $R_{u,v} = R_{v,u}$ .

For formation control problems the first assumption is not restrictive since the desired formation is usually time invariant. The second assumption says that if a measurement  $y_{u,v}$  is available to  $u$ , then the measurement  $y_{v,u}$  is available to  $v$ , although both measurements will be corrupted with noise. Since the noise corrupting the measurement of  $x_u - x_v$  available to  $u$  will be (in general) different from

the noise on the measurement of  $x_v - x_u$  available to  $v$ , these two measurements are distinct. The assumption that the noises affecting the measurements on the edge  $(u, v)$  has the same auto-covariance as the noise on  $(v, u)$  is likely to be satisfied when they use similar sensors. The assumption that whenever  $(u, v)$  exists,  $(v, u)$  also exists may fail to hold in certain situations, such as when one node's sensor fails. We will refer to the assumption that one-way, asymmetric measurement never takes place, together with the assumption that the noise covariances on parallel edges are equal, as *symmetric measurement*. Fig. 6.1 shows an example of such a symmetric directed graph. In the terminology of Section 2.2.2,  $\mathcal{G}$  is a matrix-weighted graph with edge weights specified by a function  $W : \mathcal{E} \rightarrow \mathbb{S}^{k+}$ , where  $W_{u,v} = R_{u,v}^{-1}$ .

Our goal is to determine how the structure of the graph and the measurement noises affects the formation error. The formation error of a node  $u \in \mathcal{V}$  is defined as

$$e_u(t) = x_u(t) - x_o(t) - r_{u,o}. \quad (6.4)$$

where  $r_{u,o}$  is the desired relative position of  $u$  w.r.t. the reference  $o$ . Due to measurement noise, this error will be random, so we also look at its auto-covariance:

$$\Sigma_{u,o} = \mathbb{E} [(e_u(t) - \mathbb{E}[e_u(t)])(e_u(t) - \mathbb{E}[e_u(t)])^T]. \quad (6.5)$$

We use a control law in which each node uses all its measurements to construct an optimal estimate of the difference between its currently position and what this “should” be, in view of what it know about its neighbors positions. The measurements available to an arbitrary node  $u \in \mathcal{V}$  are

$$y_{u,v} = x_u - x_v + \epsilon_{u,v}, \quad \forall v \in \mathcal{N}_u,$$

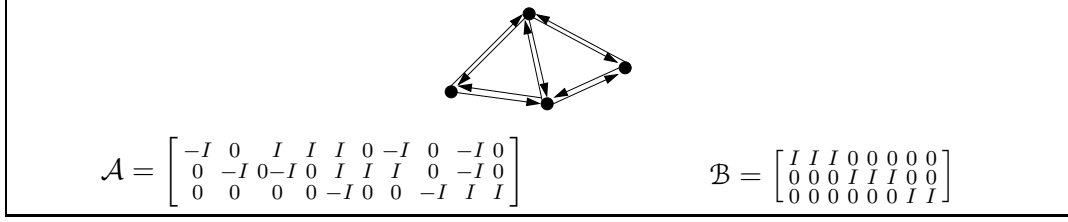


Figure 6.1. A symmetric interconnection graph, its generalized incidence matrix  $\mathcal{A}$ , and the matrix  $\mathcal{B}$ .

where  $\mathcal{N}_u \subset \mathcal{V}$  denotes set of nodes  $v$  such that  $(u, v) \in \mathcal{E}$ . If node  $u$  assumes that all its neighbors are correctly positioned then, according to (6.1), the desired position of  $u$  is given by any one of the following equations

$$x_u^d = x_v + r_{u,v}, \quad \forall v \in \mathcal{N}_u.$$

Combining the two previous sets of equations, we obtain

$$y_{u,v} = x_u - x_u^d + r_{u,v} + \epsilon_{u,v}, \quad \forall v \in \mathcal{N}_u,$$

from which nodes  $u$  estimates its position error  $x_u - x_u^d$ . It is straightforward to show that the best linear unbiased estimate of  $x_u - x_u^d$  is given by

$$D_u^{-1} \sum_{v \in \mathcal{N}_u} R_{u,v}^{-1} (y_{u,v} - r_{u,v}),$$

where  $D_u := \sum_{v \in \mathcal{N}_u} R_{u,v}^{-1}$ . This motivates the following negative proportional control law for the nodes

$$\dot{x}_u = -\gamma D_u^{-1} \sum_{v \in \mathcal{N}_u} R_{u,v}^{-1} (y_{u,v} - r_{u,v}), \quad \forall u \in \mathcal{V} \setminus \{o\}, \quad (6.6)$$

where  $\gamma$  denotes some positive number.

For analysis purposes it is convenient to describe the system dynamics in term of positions with respect to the reference. Defining  $\tilde{x}_u = x_u - x_o$ , one concludes

that

$$\dot{\tilde{x}}_u = -\gamma D_u^{-1} \sum_{v \in \mathcal{N}_u} R_{u,v}^{-1} (\tilde{x}_u - \tilde{x}_v - r_{u,v} + \epsilon_{u,v}) - \dot{x}_o,$$

for every  $u \in \mathcal{V} \setminus \{o\}$ . By stacking all the positions  $\tilde{x}_u$ ,  $u \in \mathcal{V} \setminus \{o\}$  in a column vector  $\tilde{\mathbf{x}}$ , the above systems can be written as follows:

$$\dot{\tilde{\mathbf{x}}} = -\gamma \mathcal{M}^{-1} \mathcal{L} \tilde{\mathbf{x}} + \gamma \mathcal{M}^{-1} \mathcal{B} \mathcal{W} (r - \epsilon) - \dot{x}_o \mathbf{1}, \quad (6.7)$$

where  $r$  is a column vector obtained by stacking all the  $r_{u,v}$  on top of each other;  $\epsilon$  is a column vector obtained by stacking all the  $\epsilon_{u,v}$ ;  $\mathbf{1}$  is a  $n - 1 \times 1$  column vector of all 1's;  $\mathcal{W} > 0$  is a block-diagonal matrix with  $k$  rows/columns for each edge in  $\mathcal{E}$ , with the weights  $W_{u,v} := R_{u,v}^{-1}$ ,  $(u, v) \in \mathcal{E}$  in the diagonal;  $\mathcal{M} > 0$  is a block-diagonal matrix with  $k$  rows/columns for each node in  $\mathcal{V} \setminus \{o\}$ , with  $D_u$ ,  $u \in \mathcal{V} \setminus \{o\}$  as defined earlier in the diagonal;  $\mathcal{L} = \frac{1}{2} \mathcal{A}_b \mathcal{W} \mathcal{A}_b^T$  where  $\mathcal{A}_b$  is the generalized basis incidence matrix for the directed graph  $(\mathcal{V}, \mathcal{E})$ ; and  $\mathcal{B}$  is a matrix with  $k$  rows for each vertex in  $\mathcal{V} \setminus \{o\}$  and  $k$  columns for each edge in  $\mathcal{E}$ , constructed as follows: the  $k$  columns corresponding to edge  $(u, v) \in \mathcal{E}$  are all equal to zero except for the block corresponding to the node  $u$ , which is equal to  $I_k$ . The white noise process  $\epsilon$  has block diagonal auto-covariance matrix given by  $E[\epsilon(t_1)\epsilon^T(t_2)] = \delta(t_1 - t_2)\mathcal{W}^{-1}$ . Figure 6.1 shows an example of the matrices defined above.

The compact form of the closed loop dynamics (6.7) explains the terminology Laplacian disagreement control – because of the appearance of the Dirichlet Laplacian  $\mathcal{L}$ . Similar control laws have been studied in the literature on multi-vehicle control and multi-agent consensus [29, 30, 128].

The main result of this section is the following:

**Theorem 6.3.1.** *Consider the problem of formation control with a finite number of mobile nodes, described by the directed graph  $\mathcal{G}$  and a function  $R : \mathcal{E} \rightarrow \mathbb{S}^{k+}$*

that describes the auto-covariance of the measurement error process (6.3), in which every node implements the Laplacian disagreement control law (6.6). When assumption 6.3.1 holds, the closed loop is stable irrespective of the number of vehicles, and the steady state covariance matrices for the formation error for node  $u$  is given by

$$\Sigma_{u,o} = \gamma R_{u,o}^{\text{eff}},$$

where  $R_{u,o}^{\text{eff}}$  is the matrix-valued effective resistance between  $u$  and  $o$  in the generalized electrical network  $(\mathcal{G}, R)$ .  $\square$

The scaling of matrix-valued effective resistance  $R_{u,o}^{\text{eff}}$  as a function of distance  $d_{u,o}$  of  $u$  from the reference  $o$  determine how the structure of the graph  $\mathcal{G}$  affects the growth of the effective resistance, and therefore formation error covariance. Effective resistances in graphs were studied extensively in Chapter 5, as it was identified that the structure of the graph greatly affects the effective resistances. In sparse graphs, the effective resistance grows fast with distance from the reference, whereas in dense networks it grows slowly. In view of Theorem 6.3.1, this dependence of effective resistance on graph structure has significant implications for the problem of formation control, which are discussed below.

### 6.3.1 Implications for man-made autonomous agents

The preceding discussion shows that the maximum tracking errors in two networks consisting of the same number of agents can be quite different. As a result, some networks are more scalable than others in terms of tracking performance. This knowledge can be used for designing networks that are formed by groups of mobile autonomous agents, such as UAVs. Frequently, the formation structure of

such agents is designed solely on the basis of the task that the group is expected to perform. However, our results show that a formation structure itself imposes fundamental limitations on how well that formation can be maintained by the agents. Thus, if the agents are required to maintain their formation accurately, then the desired formation itself has to be appropriately chosen. For example, it will be unwise to ask a large group of agents to fly in a single line while maintaining very accurate spacings between neighbors, since we know that in such a graph the tracking error grows linearly with the number of agents.

### 6.3.2 Implications for Swarming in Nature

While the exact nature of motion coordination among biological agents is still a mystery, the control law (6.6) is nevertheless an approximation of the motion coordination schemes that are proposed to explain swarming behavior in animals [135, 136]. This control law is extremely simple and requires only information about nearby agents, which can be obtained by animals through their vision and/or auditory sensors. Moreover, measurement noise is likely to affect the relative position estimates as modelled in (6.2).

The analogy between effective resistance and formation error covariance can explain a number of puzzling observations from nature. For example, it is well known that several species of birds fly in a “V”-formation (cf. Fig. 6.2(a)). Although why this happens is still a matter of debate (both drag reduction and better visual cue have been offered to explain this phenomenon; see [22] for arguments for the former and [127] for arguments against the former and for the latter), it is observed that the birds close to the leader can maintain their relative positions better than the birds toward the end of the arms of the “V”. An example is shown in the



(a) A flock of birds in “V”-formation



(b) A school of fish

Figure 6.2. Examples of 1-D and 3-D network topologies in natural swarms. Photograph in (b) courtesy Sergey Parinov (<http://www.sergeyphoto.com>)

photograph of Figure 6.2(a). Such large formation error in bird flocks might be explained by the fact that a V-formation is sparse in 1-dimension and hence the effective resistance are large when the number of birds in the flock is large (see Theorem 5.3.1). On the other extreme, schools consisting of millions of fish are known to move together in a 3-dimensional structure in a surprisingly agile fashion [137]. In a large school of fish such as the one shown in Fig. 6.2(b), the interconnection topology is 3-dimensional. It is not hard to see that the network in such a large school will be dense in 3-D, if we image it as being a part of an infinite graph. Hence the tracking error variance of the individuals remain bounded even when the number of nodes (fish) making up the school is arbitrarily large. This might explain why large fish schools can move together and maneuver quickly even while forming an extremely large network while a comparatively small number of birds flying straight find it difficult to keep a constant separation.

*Proof of Theorem 6.3.1.* Construct a graph  $\bar{\mathcal{G}} = (\mathcal{V}, \bar{\mathcal{E}})$  whose edge set  $\bar{\mathcal{E}}$  consists of exactly one for every pair of parallel edges in  $\mathcal{E}$ . Construct an edge-weight function  $\bar{W} : \bar{\mathcal{E}} \rightarrow \mathbb{S}^{k+}$  as follows:

$$\bar{W}_{u,v} = 2W_{u,v} \quad (6.8)$$

It can be verified that due to the assumption of symmetry,  $\mathcal{L} = \bar{\mathcal{A}}_b^T \bar{\mathcal{W}} \bar{\mathcal{A}}_b^T$ , where  $\bar{\mathcal{A}}_b$  is the basis incidence matrix of  $\bar{\mathcal{G}}$  w.r.t.  $o$  and  $\bar{\mathcal{W}}$  is a block diagonal matrix with  $\bar{W}_e, e \in \bar{\mathcal{E}}$  on the diagonal. Recalling the definition of a Dirichlet Laplacian of a matrix weighted graph from Section 2.2.2, we see that  $\mathcal{L}$  is exactly the matrix-weighted Dirichlet Laplacian for the matrix-weighted graph  $\bar{\mathcal{G}} = (\mathcal{V}, \bar{\mathcal{E}})$  with boundary  $\{o\}$  and with weight  $\bar{W}_{u,v} = R_{u,v}^{-1}$  on every edge  $(u, v)$ . Since  $\gamma > 0$ , and  $\mathcal{M}$  and  $\mathcal{L}$  are positive definite (see Theorem 2.2.1), we conclude that (6.7) is an asymptotically stable system.

We further re-write (6.7) as

$$\dot{\tilde{\mathbf{x}}} = -\gamma \mathcal{M}^{-1} \mathcal{L} \tilde{\mathbf{x}} + w + b,$$

where  $b := \gamma \mathcal{M}^{-1} \mathcal{B} \mathcal{W} r - \dot{x}_o \mathbf{1}$  and  $w := -\gamma \mathcal{M}^{-1} \mathcal{B} \mathcal{W} \epsilon$  is a white noise random process with auto-covariance matrix given by

$$\begin{aligned} \mathbb{E}[w(t_1)w^T(t_2)] &= \gamma^2 \mathcal{M}^{-1} \mathcal{B} \mathcal{W} \mathbb{E}[\epsilon(t_1)\epsilon^T(t_2)] \mathcal{W} \mathcal{B}^T \mathcal{M}^{-1} \\ &= \gamma^2 \delta(t_1 - t_2) \mathcal{M}^{-1} \mathcal{B} \mathcal{W} \mathcal{B}^T \mathcal{M}^{-1} = \gamma^2 \delta(t_1 - t_2) \mathcal{M}^{-1}, \end{aligned}$$

where we used the fact that  $\mathcal{B} \mathcal{W} \mathcal{B}^T = \mathcal{M}$ . Since the Lyapunov equation

$$-\gamma \mathcal{M}^{-1} \mathcal{L} \Sigma_\infty - \gamma \Sigma_\infty \mathcal{L} \mathcal{M}^{-1} + \gamma^2 \mathcal{M}^{-1} = 0$$

has a positive definite solution

$$\Sigma_\infty = \frac{\gamma}{2} \mathcal{L}^{-1},$$

it is straightforward to show that the covariance matrix of  $\tilde{x}$  converges to  $\Sigma_\infty$ . In particular, the steady-state covariance matrix of the relative position  $\tilde{x}_u := x_u - x_o$  is given by  $k \times k$  diagonal block of  $\Sigma_\infty$ .

From Corollary 4.6.1, this diagonal block is  $\gamma/2$  times the effective resistance  $R_{u,o}^{\text{eff}}$  between  $u$  and  $o$  in the generalized electrical network  $(\bar{\mathcal{G}}, \bar{R})$  where  $\bar{R}_{u,v} = \bar{W}_{u,v}$ ,  $(u, v) \in \bar{\mathcal{E}}$ . Because of the role of parallel resistance formula (see Proposition 4.6.1 and Section 2.2.3), this effective resistance in the network  $(\bar{\mathcal{G}}, \bar{R})$  is twice the effective resistance in the network  $(G, R)$ , which proves the theorem. ■

## 6.4 Lower bound on the Dirichlet Laplacian spectrum from effective resistance

In this section we will establish bounds on the smallest eigenvalue of the Dirichlet Laplacian using effective resistances. Motivation for obtaining bounds on this eigenvalue is summarized in the next section.

### 6.4.1 Role of the Dirichlet Laplacian spectrum

#### 6.4.1.1 Convergence rate of discrete-time algorithms

We have already seen in Chapter 3 that the convergence rate of a distributed estimation algorithm depends on this eigenvalue (cf. Theorem 3.3.4). The number of iterations  $n_{\text{iter}}(\epsilon)$  needed to drive the error ratio below a certain value  $0 < \epsilon < 1$  is given by

$$n_{\text{iter}}(\epsilon) = \Theta\left(\frac{1}{\lambda_{\min}(L)}\right),$$

where  $L$  is the Dirichlet Laplacian. Similar results hold for the well-known average consensus algorithms, in which the nodes of a multi-agent network update their state by computing the average of their states with that of their neighbors [138]. When the communication graph of the average consensus algorithm is a fixed undirected graph  $\mathcal{G}$ , and the nodes run the average consensus algorithm to reach consensus with a reference node, which keeps its state fixed, then the error dynamics of the nodes can be expressed as

$$x^{(i+1)} = Jx^{(i)},$$

where  $J$  is the Jacobi iteration matrix for the graph  $\mathcal{G}$  with a reference nodes  $o$ , which was defined earlier in 3.19. Since this is identical to the error dynamics of the Jacobi algorithm described in Section 3.3.1.2, the results of Theorem 3.3.4 apply. So the convergence rates of both the Jacobi and average consensus algorithms can be directly obtained from  $\lambda_{\min}(L)$ .

#### 6.4.1.2 Convergence rate of continuous-time algorithms

Recalling the formation control example of Section 6.3, we see that the stability of the closed loop is determined by that of the system

$$\dot{\mathbf{x}} = -\gamma D^{-1}L\mathbf{x}. \quad (6.9)$$

Therefore, the time constant of formation errors is given by the smallest eigenvalue of the matrix  $D^{-1}L$ , which is given by  $\frac{1}{d_{\max}}\lambda_{\min}(L)$ . The dynamics (6.9) also represent a continuous time consensus algorithm [29], where nodes of the network are trying to reach consensus with the reference  $o$ . Therefore the time constants of such consensus algorithms are also dependent on  $\lambda_{\min}$ .

Apart from the examples above, we will see in Section 7.3.1 another example of  $\lambda_{\min}$ 's role, in case of control of a platoon with a dynamic compensator.

### 6.4.2 Effective resistance between a node and a set of nodes

Before proceeding further, we define the matrix-valued effective resistance between a node and a set of nodes, which can be thought of as an extension of the generalized effective resistance between two nodes defined in Section 4.4.2. This effective resistance definition is useful when there are more than one reference nodes, e.g., when there are multiple leaders in a formation.

Consider a directed graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  and let  $\mathcal{V}_r \subset \mathcal{V}$ . Recall from Section 2.2.1 that  $\mathcal{G}$  is *weakly connected to*  $\mathcal{V}_r$ , where if there is a (undirected) path from every node in the graph to at least one of the nodes in  $\mathcal{V}_r$ . Theorem 2.2.1 shows that the Dirichlet Laplacian  $\mathcal{L}$  of the network  $(\mathcal{G}, R)$  is invertible if and only if  $\mathcal{G}$  is weakly connected to  $\mathcal{V}_r$ . Recall that  $\mathcal{L} = \mathcal{A}_b \mathcal{W} \mathcal{A}_b^T$ , where  $\mathcal{A}_b$  is the generalized basis incidence matrix of  $\mathcal{G}$  and  $\mathcal{W}$  is the block diagonal weight matrix:  $\mathcal{W} := \text{diag}(W_1, \dots, W_m)$ , where  $m$  is the number of edges (see Section 2.2.2).

We now formally define a *node*  $u$ 's *effective resistance to*  $\mathcal{V}_r$ , denoted by  $R_u^{\text{eff}}(\mathcal{V}_r)$ , as the  $k \times k$  block in the main diagonal of  $\mathcal{L}^{-1}$  corresponding to the  $k$  rows/columns associated with the node  $u \in \mathcal{V} \setminus \mathcal{V}_r$ . This terminology is justified by the fact that the matrix  $R_u^{\text{eff}}(\mathcal{V}_r)$  also express a map from (matrix-valued) currents to (matrix-valued) voltages in an appropriately defined electrical network, especially when the reference node set consist of a single node (see Section 4.4.2). Here we restrict ourselves to finite networks, unlike Chapter 4.

### 6.4.3 Lower bound from effective resistance

The lower bound on the smallest eigenvalue of the Dirichlet Laplacian follows from the next lemma.  $\mathcal{L}$  is the Laplacian and  $\mathcal{L}$  is the Dirichlet Laplacian.

**Lemma 6.4.1 (Spectrum of  $\mathcal{L}$  and  $\mathcal{L}$ ).** *Assume that  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  is weakly connected to  $\mathcal{V}_r \subset \mathcal{V}$  and denote by  $\lambda_1(\mathcal{L}) \leq \lambda_2(\mathcal{L}) \leq \dots \leq \lambda_{nk}(\mathcal{L})$  the sorted eigenvalues of  $\mathcal{L}$  and by  $\lambda_1(\mathcal{L}) \leq \lambda_2(\mathcal{L}) \leq \dots \leq \lambda_{(n-n_r)k}(\mathcal{L})$  the sorted eigenvalues of  $\mathcal{L}$ . For every  $i \in \{1, 2, \dots, (n - n_r)k\}$*

$$\lambda_i(\mathcal{L}) \geq \frac{1}{\sum_{u \in \mathcal{V} \setminus \mathcal{V}_r} \text{trace } R_u^{\text{eff}}(\mathcal{V}_r)} > 0 \quad (6.10)$$

and

$$\lambda_i(\mathcal{L}) \leq \lambda_i(\mathcal{L}) \leq \lambda_{i+kn_r}(\mathcal{L}). \quad (6.11)$$

□

*Proof of Lemma 6.4.1.* By definition, the effective resistance  $R_u^{\text{eff}}(\mathcal{V}_r)$  is a diagonal block of  $\mathcal{L}^{-1}$ . The inequality (6.10) is a consequence of the fact that any eigenvalue of the positive definite matrix  $\mathcal{L}^{-1}$  can be upper-bounded by its trace, which can be obtained by adding up all the traces of its diagonal blocks  $R_u^{\text{eff}}(\mathcal{V}_r)$ ,  $u \in \mathcal{V} \setminus \mathcal{V}_r$ . This means that every eigenvalue of  $\mathcal{L}^{-1}$  satisfies:

$$\lambda_i(\mathcal{L}^{-1}) \leq \text{trace } \mathcal{L}^{-1} = \sum_{u \in \mathcal{V} \setminus \mathcal{V}_r} \text{trace } R_u^{\text{eff}}(\mathcal{V}_r),$$

from which (6.10) follows since the eigenvalues of  $\mathcal{L}^{-1}$  and  $\mathcal{L}$  are reciprocals of each other. The inequality (6.11) is a direct application of the Interlacing Eigenvalues Theorem [139, Theorem 4.3.15] to the symmetric matrices  $\mathcal{L}$  and  $\mathcal{L}$ , upon noting that  $\mathcal{L}$  is a principal submatrix of  $\mathcal{L}$  (cf. definitions of  $\mathcal{L}$  and  $\mathcal{L}$  in Section 2.2.2). ■

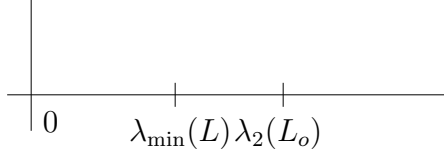


Figure 6.3. The relationship between algebraic connectivity and the smallest eigenvalue of the Dirichlet Laplacian.

To understand the implication of this result and its use, consider the case when all the edge weights are scalars and equal to unity, i.e.,  $W_e = 1$  for every  $e \in \mathcal{E}$ . In this case  $\mathcal{L} = L_o$ , where  $L_o$  is usual graph Laplacian of  $\mathcal{G}$ , thinking of  $\mathcal{G}$  as an undirected graph. The unweighted Dirichlet Laplacian  $L$  with the boundary  $\{o\}$  is a principal submatrix of  $L_o$  obtained by removing the row and column corresponding to  $o$ . By the interlacing of eigenvalues for symmetric matrices, we have

$$\lambda_{\min}(L) \leq \lambda_2(L_o),$$

where  $\lambda_2(L_o)$  is the second smallest eigenvalue of the Laplacian, which is also called the *algebraic connectivity* of  $\mathcal{G}$  [82]. This interlacing is shown graphically in Figure 6.3.

Therefore any upper bound on the algebraic connectivity is an upper bound on  $\lambda_{\min}(L)$  as well, but a lower bound on the algebraic connectivity is not necessarily a bound on  $\lambda_{\min}$  of any kind. As a result, although an extensive literature exists on bounding the algebraic connectivity of a graph (see [140, 141], and especially [81] for a good overview), these results are not useful in bounding  $\lambda_{\min}$  from below. On the other hand, we need lower bounds on  $\lambda_{\min}$  to bound the worst case performance for the applications discussed in Section 6.4.1.

## 6.5 Comments and open problems

In this section we saw evidence that the matrix valued effective resistance introduced in earlier chapters is useful in the analysis of decentralized control problems as well. The effect of graph structure in propagating noise was examined, and it was shown that the noise propagation can be characterized by the effective resistance. Effective resistance also yielded an unexpected benefit - a lower bound on the Dirichlet Laplacian eigenvalues was obtained as a function of the effective resistances. This bound is potentially valuable since there are few results on lower bounding Dirichlet Laplacian eigenvalues, which appear in several control and estimation problems.

To make good use of this bound requires knowledge of the effective resistances in the graph. Effective resistances have seen renewed popularity in recent years. Several aspects of effective resistances in graphs have been investigated. After the seminal work of [2] on transience and recurrence of random walks in infinite graphs using effective resistance, Chandra *et al.* [14] showed that the cover and commute times of a random walker in a finite graph is also captured by the effective resistances in the graph. The sum of all pairwise effective resistance for a number of special graphs have been studied in Zhang and Yang [142], whereas Ghosh *et al.* [143] provides numerical methods for minimizing the total effective resistance by choosing the edge resistances appropriately (subject to some constraints). Wu [120] obtained exact formulas for effective resistances in finite 2D and 3D grids. The matrix-valued effective resistances when each edge has a constant generalized resistance, can be easily obtained from these results by employing Lemma 4.6.1 that relates the two.

Only time-invariant interconnections were examined in regards to both error

propagation and stability margin. In certain situations the interconnections between the agents might vary with time. Error propagation in such time-varying graphs needs is an interesting open problem. Furthermore, the interconnection structure considered was symmetric, which is the reason the Dirichlet Laplacian and effective resistance appears naturally. Revisiting the issues examined here, but with asymmetric interconnection, is a wide open problem.

Another limitation of the formation control problem described in Section 6.3 is that the agent dynamics and the control laws are quite simple, with agent dynamics assumed to be first order and control law, proportional. There is a need to examine the issues of error propagation in multi-agent control systems with more complex dynamics and control laws. One such problem will be examined in the next chapter.

# Chapter 7

## Control of vehicular platoons: symmetric bidirectional control

### 7.1 Introduction

In this chapter we study the problem of controlling a string of vehicles moving in one dimension such that they all follow a lead vehicle with a constant spacing between successive vehicles (c.f. Figure 7.1). The capacity of highways can be increased by a significant amount if small inter-vehicular distances can be maintained [6]. Since human drivers cannot be expected to maintain small inter-vehicular distances due to safety reasons, one way to achieve this objective is automated driving using feedback control. A successful demonstration of a platoon of eight vehicles automatically controlled to follow a lead vehicle was conducted in 1997 by the National Automated Highway Systems Consortium under the California PATH (Partners for Advance Transit and Highways) program [144]. Due to its relevance to developing automated highway systems, the problem of

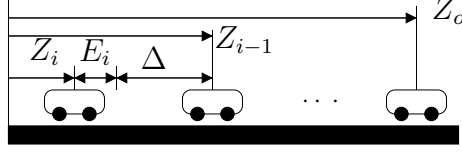


Figure 7.1. platoon of vehicles

controlling a platoon of vehicles has been studied extensively. We will discuss the prior work done in this problem in Section 7.2.1.

The automated platoon problem is a special case of the formation control problem that was already introduced in the previous chapter. The difference here is that the formation we are trying to keep is in  $\mathbb{R}^1$  instead of being in 2 or 3 dimensions. Simplification in the spatial dimension is accompanied by a complexification in other aspect of the problem – we will allow higher order dynamics and controllers than in Chapter 6.

Consider a platoon of  $N$  vehicles moving in one dimension following a lead vehicle, as shown in Figure 7.1. The lead vehicle moves independently of the other  $N$  vehicles, who try to maintain a constant gap  $\Delta$  between successive vehicles. The lead vehicle may be a real vehicle with its own dynamics, or a fictitious vehicle that represents a reference trajectory provided to the first vehicle of the platoon. Let  $Z_0(t)$  denote the position of the lead vehicle and  $Z_i(t)$ ,  $i \in \{1, 2, \dots, N\}$ , the position of the  $i^{\text{th}}$  vehicle. The  $i^{\text{th}}$  vehicle can measure the errors with respect to its predecessor and follower, namely,  $\zeta_{i,i-1}(t)$  and  $\zeta_{i+1,i}(t)$ , by on-board sensors such as radars, where

$$\zeta_{i,i-1}(t) := (Z_{i-1}(t) - Z_i(t) - \Delta) + \epsilon_{i,i-1}(t),$$

where the desired spacing,  $\Delta$ , is a positive constant and  $\epsilon_{i,i-1}(t)$  is measurement noise.

Our interest is in decentralized control architectures in which every vehicle computes its control signal based on locally available spacing error measurements. The control refers to the signal fed to the actuator that drives the vehicle. How the control signal affects the position and velocity of the vehicle therefore depends on the model of the vehicle dynamics. Although the dynamics of a highway vehicle are typically non-linear, they can be converted to that of a double integrator (i.e., a point mass without damping) by feedback linearization [145, 146]. However, since the dominant aerodynamic drag on a vehicle is quadratic in its velocity (see the models described in [146, 147]) a Jacobian linearization around the nominal velocity results in a model that has a single integrator in series with a low pass filter. We will consider both types of linear models of vehicles, with two integrators as well as one.

An extensively studied decentralized control architecture is the *predecessor following*, in which the control action on a particular vehicle depends on its spacing error with the predecessor, i.e., the vehicle in front of it. However, the predecessor following architecture is known to suffer from the limitation that the disturbances acting on the vehicles lead to large inter-vehicular spacing errors. We will discuss these limitations in Section 7.2.

Another decentralized control architecture that is investigated in the literature is *bidirectional control*. In this scheme, the control action on a particular vehicle depends on the spacing errors with respect to its predecessor and its follower. Most human drivers use information about preceding and following vehicles to control their own vehicles - especially in heavy traffic, so bidirectional control is intuitively appealing. In *symmetric bidirectional control*, the control effort is equally dependent on the spacing errors with the preceding vehicle and the following vehicle. The effect of disturbances acting on the vehicles on the spacing

errors with symmetric bidirectional control was analyzed by Seiler *et al.* [32], who showed that when the vehicle model has two integrators and controller does not have an integrator, the disturbances acting on the vehicles result in large spacing errors.

In this chapter we examine symmetric bidirectional control and answer the questions left unanswered in [32]. We answer the question of stability and disturbance amplification when the vehicle model has either one, or more than two integrators; and also characterize the effect of the lead vehicle's trajectory on the spacing errors of all the vehicles. The results are independent of the choice of the controller but are due to the interconnection structure imposed by symmetric bidirectional architecture.

*Chapter organization:* We first review the literature on the control of platoons in Section 7.2 and then briefly summarize our results. In Section 7.2, we formulate the problem and present the results in Section 7.3. Every theorem is about a different aspect of the problem, such as stability, steady-state tracking error, and disturbance amplification, and hence is presented in a separate subsection. An intuitive explanation of the results is provided in Section 7.3.5. The chapter concludes with comments on open issues in Section 7.4.

## 7.2 Prior work and contributions

### 7.2.1 Prior work on vehicular platoons

There is an extensive literature on automated vehicular platoon problem. Early work on platoons can be dated back at least half-a-century ago, to [31]. Early interest, in the 50's and 60's, in this problem stemmed from the proposals to

build Automated Guided Transit (AGT) system with electrically powered vehicles as a way of mitigating increasing urban problem of “..congested roadways, large numbers of accidents and fatalities, and extremely powerful automobiles [148]”. Due to space limitations, we will only review work on the so-called “constant-spacing policy”, in which the goal is to maintain constant inter-vehicular separation. Other policies, such as constant-time headway and constant safety-factor policies [149] will not be discussed.

Among decentralized schemes, the predecessor-following architectures was the one studied the first and perhaps the most (see [7, 149–151] and references therein). Still, it was recognized early on that disturbances acting on the vehicles tend to get amplified in predecessor-following control. For this reason, predecessor-following was considered not stable in [31]. The disturbance amplification tendency is usually referred to as “string instability”. Although a precise definition of string instability – mainly motivated by the platoon problem – was offered much later in [152], the term itself was in vigorous use for a long time. Mention of the phrase can be found in such early references as [149, 151, 151]. The term “spatial asymptotic stability” was also used in place of string stability in [150]. More recently, the term “slinky-type effects” has also been used to describe the phenomenon of error amplification in vehicle strings [153].

Although extensively studied, it took quite a while before a thorough understanding of the limitations of the predecessor-following architecture’s was developed. That the limitations are fundamental in nature and independent of the controller’s design, was shown much later, by Seiler *et al.* [32].

As the limitations of predecessor-following were observed quite early, it led to the proposal of a *predecessor-and-leader-following* architecture, in which the control action at a vehicle depends on, in addition to the predecessor’s error, the

error with respect to the lead vehicle in the platoon. It was shown, e.g., in [151], that such a scheme will in fact damp out disturbances all along the platoon. Clearly, it is not a decentralized architecture. The demonstration of automated platooning in 1997 used this architecture [144].

Unfortunately, the predecessor-and-leader-following scheme also suffers from severe limitations. In particular, it was shown in Liu *et al.* [154] the closed loop becomes highly sensitive to the time delays incurred in transmitting the lead vehicle's position information to the rest of the platoon.

The LQR control of platoons, which typically leads to a centralized architecture, was investigated as early as 1966 [155]. LQR control of an infinite string of vehicles were investigated in [156] and in [157], though a more complete analysis was provided only in 2005 by Jovanović and Bamieh [158]. It was shown in [158] that the optimal control of the platoon is effectively ill-posed when the number of vehicles is large, namely, that the time constant of the closed loop become arbitrarily small as the number of vehicles become arbitrarily large. Since the optimal control of a platoon suffers from such limitations, it is perhaps not surprising that the decentralized control suffers from limitations as well.

The discussion above shows that even the centralized schemes, such as LQR and leader-and-predecessor-following fail to remove the difficulties of decentralized predecessor-following scheme. Therefore, it behooves us to examine other decentralized architectures that, although cannot be expected to not suffer from any of the limitations discussed above, at least has the potential of performing reasonably well. The bidirectional architecture is a natural choice, which was analyzed in [159] and claimed to not suffer from string instability. This claim was, however, erroneous since the measure of string stability used only ensured that disturbances will be damped out in going from front to the back of the platoon, but not the

other way around. Bidirectional architecture was also investigated in the non-linear setting by Zhang *et al.* [153], whose proposed design was able to achieve stability without “slinky-type effects”, which is another name preferred by several researchers in place of string instability. In the linear case, the disturbance amplification properties of the symmetric bidirectional scheme were examined in [32], which showed that for a certain class of plants and controllers this architecture also suffers from limitations that are controller-independent.

### 7.2.2 Main results

The results presented in this chapter are summarized below. All the results apply only to the case of symmetric bidirectional architecture, and when the vehicle and controller models are linear. In the sequel, the plant  $H(s)$  denotes the transfer function from control input to vehicle position, and the controller  $K(s)$  denotes the transfer function from the position error to the control input.

1. It is possible to design the controller  $K(s)$  so that the closed loop is stable for an arbitrarily large but finite number of vehicles, as long as the number of integrators in the loop  $H(s)K(s)$  is not more than two. When  $H(s)K(s)$  has either one or two integrators, if  $H(s)$  is non-minimum phase, for every stable controller  $K(s)$ , the closed loop will become unstable for a sufficiently large number of vehicles. If the total number of integrators in  $H(s)K(s)$  is more than two, then the closed loop will be unstable for a sufficiently large number of vehicles, irrespective of how  $K(s)$  is designed.
2. When  $H(s)K(s)$  has two integrators, if the lead vehicle moves at constant velocity, the steady state spacing errors for every vehicle will go to 0, irrespective of the number of vehicles in the platoon. When  $H(s)K(s)$  has only

one integrator, if the lead vehicle moves at a constant velocity, the steady state error is finite for a finite platoon size, but the norm of this error grows without bound as the number of vehicles in the platoon increases.

3. When  $H(s)K(s)$  has two integrators, in the absence of disturbances on the vehicles, if the lead vehicle trajectory deviates from a constant-velocity one, the  $\mathcal{L}_2$  norm of the spacing errors will grow unbounded as the number of vehicles increases, even if the deviation has bounded  $\mathcal{L}_2$ -norm. However, when  $H(s)K(s)$  has only one integrator, if the deviation of the lead vehicle's trajectory from a constant velocity one is  $\mathcal{L}_2$ -norm bounded, then the spacing errors of the entire platoon are  $\mathcal{L}_2$  norm bounded, too, irrespective of the number of vehicles.
4. When  $K(s)$  has no integrators, the  $\mathcal{H}_\infty$  norm of the transfer function from the disturbances acting on the vehicles to the spacing errors will grow without bound as the number of vehicles increases, irrespective of whether  $H(s)$  has either one or two integrators. Thus, even if the lead vehicle is moving at constant velocity, if disturbances are present in the control signal – as they invariably will – large spacing errors will result for a large platoon.

The case of  $H(s)K(s)$  having no integrators is not considered, since a realistic model of a vehicle for a highway will have at least one integrator. In fact, vehicle models as simply double integrators (fully actuated point masses with no damping) are quite common in the literature [32, 145, 146]. Even models with three integrators have been studied in the literature, e.g., the vehicle model in [147], which result from feedback linearization of high-order non-linear vehicle models. Effect of disturbance when  $K(s)$  and  $H(s)$  each has one integrator has not been studied in this dissertation.

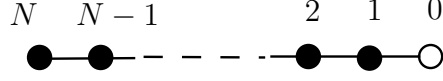


Figure 7.2. The interconnection architecture graph for symmetric bidirectional control. The symmetry of the interconnection graph is manifested in the graph being undirected.

It is important to notice that the results are independent of how the controller  $K(s)$  is chosen. In short, the limitations of the symmetric control architecture cannot be ameliorated with clever control design. The results in this chapter have been reported earlier in [160]. The results for  $H(s)$  with two or more integrators were also proved by Yadlapalli *et al.* [161] independently.

### 7.3 Problem statement and main results

Let  $\mathbb{N}$ ,  $\mathbb{R}$  and  $\mathbb{C}$  denote the set of natural, real and complex numbers, respectively. As shown schematically in Figure 7.1, the platoon consists of  $N$  vehicles moving in one dimension following a lead vehicle, where the lead vehicle moves independently of the other  $N$  vehicles. Let  $Z_0(t)$  denote the position of the lead vehicle and  $Z_i(t)$ ,  $i \in \{1, 2, \dots, N\}$ , the position of the  $i^{\text{th}}$  vehicle. The *spacing error* of the  $i$ th vehicle is defined by

$$E_i(t) = Z_{i-1}(t) - Z_i(t) - \Delta, \quad (7.1)$$

where the desired spacing,  $\Delta$ , is a positive constant. The lead vehicle may be a real vehicle with its own dynamics, or a fictitious vehicle that represents a reference trajectory provided to the first vehicle of the platoon. The control objective is to keep the spacing error for every vehicle as small as possible while maintaining closed loop stability. We assume that,

1. the dynamics of individual vehicles are identical, and the transfer function from control input to vehicle position is denoted by  $H(s)$ ,
2.  $H(s)$  is SISO and has at least one integrator,
3. all vehicles use the same control law, and
4. the string of vehicles start with zero spacing errors, from rest, and the lead vehicle starts at  $Z_0(0) = 0$ . Hence,  $Z_i(0) = -i\Delta$ .

Let  $X(s)$  denote the Laplace transform  $\mathcal{L}(\cdot)$  of a time-domain signal  $X(t)$ :

$$Z(s) := \mathcal{L}(Z(t)).$$

Applying the assumptions, each vehicle can be modelled in the Laplace domain as

$$X_i(s) = H(s)(U_i(s) + D_i(s)) + \frac{Z_i(0)}{s}, \quad 1 \leq i \leq N, \quad (7.2)$$

where  $Z_i(0)$  is the initial position of the  $i$ th vehicle,  $U_i(s)$  is the Laplace transform of the control signal, and  $D_i(s) = \mathcal{L}(D_i(t))$  is the Laplace transform of the input disturbance  $D_i(t)$  to the  $i$ th vehicle. The effect of measurement noise can be absorbed into the input disturbances  $D_i(t)$ , so from now on we can assume that the noise free errors  $E_i(t)$  can be measured by the vehicles.

The  $i$ th spacing error in the Laplace domain is given by

$$E_i(s) = Z_{i-1}(s) - Z_i(s) - \frac{\Delta}{s}, \quad 1 \leq i \leq N. \quad (7.3)$$

Using (7.3) and (7.2) and using  $Z_i(0) = -i\Delta$ , we can write the error dynamics of the entire vehicle platoon as

$$\bar{E}(s) = Z_0(s)\phi_1 + P(s) [\bar{D}(s) + \bar{U}(s)] \quad (7.4)$$

where  $\phi_1 \in \mathbb{R}^N$  is the 1st element of the canonical basis of  $\mathbb{R}^N$  and

$$\begin{aligned}\bar{E}(s) &:= \mathcal{L}(\bar{E}(t)), \quad \bar{E}(t) = [E_1(t) \dots E_N(t)]^T, \\ \bar{D}(s) &:= \mathcal{L}(\bar{D}(t)), \quad \bar{D}(t) = [D_1(t) \dots D_N(t)]^T, \\ \bar{U}(s) &:= [U_1(s) \dots U_N(s)]^T, \\ P(s) &:= -H(s)M^T,\end{aligned}$$

where  $M$  is defined as

$$M := \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & \ddots & -1 \\ & & & 1 \end{bmatrix} \in \mathbb{R}^{N \times N}. \quad (7.5)$$

In a symmetric bidirectional control scheme, each vehicle bases its control action on the error feedback from its predecessor and follower with equal emphasis. The control action is

$$U_i(s) = K(s)(E_i(s) - E_{i+1}(s)), \quad 1 \leq i \leq N. \quad (7.6)$$

Since the last vehicle in the string does not have a follower, it uses the controller  $U_N(s) = K(s)E_N(s)$ . The vector of platoon control inputs is given by

$$\bar{U}(s) = K(s)M\bar{E}(s)$$

which is a restatement of (7.6). Eliminating  $\bar{U}(s)$  from (7.4) we can write the closed-loop error dynamics of the platoon, which is given by

$$\bar{E}(s) = G_{x_{oe}}(s)Z_o(s) + G_{de}(s)\bar{D}(s), \quad (7.7)$$

where

$$G_{x_{oe}}(s) = [I + H(s)K(s)L]^{-1}\phi_1, \quad (7.8)$$

$$G_{de}(s) = -H(s)[I + H(s)K(s)L]^{-1}M^T, \quad (7.9)$$

and  $L := M^T M \in \mathbb{R}^{N \times N}$ . The matrix  $L$  is given by

$$L = \begin{bmatrix} 1 & -1 & 0 & \dots & \\ -1 & 2 & -1 & \dots & \\ 0 & -1 & 2 & -1 & \dots \\ & & & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix}. \quad (7.10)$$

The matrix  $L$  is similar to the Laplacian matrix of the undirected graph whose nodes are the vehicles and the edges are the measurements/communications between neighboring vehicles. In fact,  $L$  is exactly the Dirichlet Laplacian (defined in Section 2.2.2) for the line graph with  $N$  nodes with unity edge weights and a node at the end as the boundary, that is shown in Figure 7.2. This graph describes the interconnection structure among the vehicles in an  $N$ -vehicle platoon with symmetric bidirectional control.

The results on various aspect of the problem, including closed loop stability, steady state error, amplification of disturbance in the lead vehicle's trajectory, and of the disturbances acting on all the vehicles, are presented next.

### 7.3.1 Closed loop stability with symmetric bidirectional control

**Theorem 7.3.1.** *Consider the closed loop error dynamics of the platoon with symmetric bidirectional control, given by (7.7).*

1. *For closed loop stability of the platoon with arbitrary  $N$ ,  $H(s)K(s)$  cannot have more than two integrators.*
2. *For closed loop stability of the platoon with  $N$  vehicles following the leader, every transfer function  $G_i(s) = 1/(1 + \lambda_i H(s)K(s))$ ,  $i = \{1, 2, \dots, N\}$  must be stable, where  $\lambda_i$  is the  $i$ th eigenvalue of the matrix  $L \in \mathbb{R}^{N \times N}$  defined in (7.10), and consequently,  $K(s)$  cannot have zeros at 0.*
3. *Define  $H(s)K(s) = C(s)/s^k$  with  $C(0)$  finite. Then for closed loop stability with arbitrary  $N$ , we must have  $C(0) > 0$ .* □

We have already discussed that for  $H(s)$  to be a reasonable model of a vehicle in a highway,  $H(s)$  must have at least one integrator. The theorem above shows that for the closed loop to be stable with symmetric bidirectional control for arbitrary number of vehicles, the vehicle dynamic model cannot have more than two integrators. Moreover, closed loop stability for arbitrary  $N$  is impossible when either the vehicle dynamics or the controller is non-minimum phase. It is clear from the Theorem that stability margin will be determined by  $\lambda_{\min}(L)$ , which is related to the effective resistances in the control architecture graph in Figure 7.2 (see Section 6.4).

### 7.3.2 Steady-state errors

In an automated highway system, it is in general desired that the vehicles move at a constant velocity for safety, comfort, and fuel-efficiency. This can be achieved by providing a constant velocity reference to the first vehicle of the platoon, which is equivalent to introducing a fictitious lead vehicle that moves at a constant velocity at all times. In this section we will show that if the lead vehicle moves at a constant velocity, i.e.,

$$Z_o(t) = Z_o^{\text{ref}}(t) = V_d t, \quad (7.11)$$

where  $V_d$  is the desired constant velocity, and  $H(s)K(s)$  has two integrators, then all the platoon spacing errors can be made to converge to 0. If  $H(s)K(s)$  has a single integrator, then the steady state platoon spacing error vector is non-zero, and the norm of the steady-state error grows without bound as  $N$  increases. This is stated in the next theorem.

**Theorem 7.3.2.** *Consider the case when there are no disturbances acting on the vehicles, i.e.,  $\bar{D}(t) \equiv 0$ , and the lead vehicle moves at a constant velocity at all*

times, i.e.,  $Z_o(t) = V_d t$ , where  $V_d > 0$  is the desired constant velocity. Let  $K(s)$  be such that it achieves closed loop stability of the platoon error dynamics with symmetric bidirectional control. Then the following are true:

1. If  $H(s)K(s)$  has two integrators, then,  $\forall N \in \mathbb{N}$ ,

$$\lim_{t \rightarrow \infty} \bar{E}(t) = 0.$$

2. If  $H(s)K(s)$  has one integrator, then for a platoon of size  $N$ ,  $\exists E_\infty \in \mathbb{R}^N$ , such that,

$$\lim_{t \rightarrow \infty} \bar{E}(t) = \bar{E}_\infty \neq 0,$$

and, for every  $R > 0$ ,  $\exists N_o \in \mathbb{N}$  such that  $\|\bar{E}_\infty\| > R$ ,  $\forall N > N_o$ , where  $\|\bar{E}_\infty\|$  denotes the Euclidean 2-norm of the  $N$ -vector  $\bar{E}_\infty$ .  $\square$

Since the steady state spacing errors grow without bound as the platoon size increases, it means for a sufficiently large platoon, there might be collisions between vehicles in the platoon.

### 7.3.3 Effect of lead vehicle's deviation from constant velocity

In certain cases, the platoon might have a lead vehicle that is not a fictitious reference but a real vehicle with dynamics. Due to disturbances entering the lead vehicle, it is reasonable to expect that the leader trajectory will deviate from the constant velocity reference trajectory, at least by a small amount. In this case, the lead vehicle's trajectory can be modelled as

$$Z_o(t) = V_d t + \zeta_o(t),$$

where  $\zeta_o(t)$  is the error from the constant-velocity trajectory.

We effect of deviations in the lead vehicles trajectory from a constant velocity one on the spacing errors in the platoon is stated in the next theorem. The proof of the result is provided in Section 7.5. In the statement of the theorem,  $\|\cdot\|$  denotes the Euclidean 2-norm of a real or complex vector and  $\|\cdot\|_\infty$  denotes the  $\mathcal{H}_\infty$ -norm of a transfer function.

**Theorem 7.3.3.** *Assume  $H(s)K(s)$  has two poles at the origin, the closed loop platoon error dynamics under symmetric bidirectional control is stable for arbitrary  $N$ . Let  $G_{x_{oe}}(s) \in \mathbb{C}^{N \times 1}$  be the transfer function from lead vehicle position  $X_o(s)$  to spacing errors  $\bar{E}(s)$  defined in (7.8). Then,*

$$\|G_{x_{oe}}\|_\infty > \beta N^{\frac{1}{2}},$$

where  $\beta$  is a constant independent of  $N$ .

Since  $G_{x_{oe}}(s)$  is also the transfer function from  $\zeta_o$  to  $\bar{E}$ , the result above shows that even if  $\|\zeta_o\|_{\mathcal{L}_2}$  is bounded,  $\|\bar{E}\|_{\mathcal{L}_2}$  will grow unbounded as  $N$  increases. The only situation when low spacing errors can be achieved with zero steady state error for all vehicles is when  $\zeta_o(t) \equiv 0$ , an unlikely scenario.

It turns out that the situation is better when  $H(s)K(s)$  has only one integrator, which is stated next.

**Theorem 7.3.4.** *Assume  $H(s)K(s)$  has one pole at the origin, the closed loop platoon error dynamics under symmetric bidirectional control is stable for arbitrary  $N$ . Let  $G_{x_{oe}}(s) \in \mathbb{C}^{N \times 1}$  be the transfer function from lead vehicle position  $X_o(s)$  to spacing errors  $\bar{E}(s)$  defined in (7.8). Then,*

$$\|G_{x_{oe}}\|_\infty < c$$

where  $c$  is a constant independent of  $N$ . □

Proof of Theorem 7.3.3 is provided in Section 7.5. A proof of Theorem 7.3.4 is not provided, since it follows directly from the arguments provided in Section 7.3.5, which offers a intuitive explanation of these results.

### 7.3.4 Disturbance propagation

To examine the effect of disturbances acting on the vehicles in the spacing errors, we have to look at the transfer function matrix from the disturbances to the spacing errors:  $G_{de}(s)$ . The question of disturbance propagation was already investigated by Seiler *et. al.* in [32], where it was shown that for the symmetric bidirectional control scheme, it is not possible to design a  $K(s)$  to achieve a uniform bound on  $\|G_{de}\|_\infty$  w.r.t.  $N$ , when  $H(s)$  has two integrators and  $K(s)$  has none. It follows from theorem 7.3.1 that if  $H(s)K(s)$  has three integrators, then the closed loop platoon error dynamics will be unstable for a sufficiently large  $N$ . This precludes the possibility of  $K(s)$  having an integrator when  $H(s)$  has two integrators. We consider only the case of  $H(s)K(s)$  having either one or two integrators. The proof is provided in Section 7.5.

**Theorem 7.3.5.** *Let the controller  $K(s)$  be such that the closed loop platoon dynamics is stable for arbitrary  $N$ . Let  $G_{de}(s) \in \mathbb{C}^{N \times N}$  be the transfer function matrix from  $\bar{D}(s)$  to  $\bar{E}(s)$  defined in (7.9). If  $K(s)$  has no integrators, then, irrespective of whether  $H(s)K(s)$  has one or two integrators,  $\|G_{de}\|_\infty > cN$  for some constant  $c$  independent of  $N$ .  $\square$*

This theorem tells us that even if the disturbances acting on the vehicles are  $\mathcal{L}_2$ -norm bounded, the  $\mathcal{L}_2$ -norm of the spacing errors due to these disturbances will grow unbounded as  $N$  grows. Therefore a symmetric bidirectional control scheme is not scalable with respect to disturbance rejection. This result was established

in [32] for vehicle models with two integrators, with the assumption that  $K(s)$  does not have any integrators. The theorem above shows that even when  $H(s)$  has only one integrator, the result in [32] holds. What happens when  $H(s)$  and  $K(s)$  each has one integrator is an open question.

### 7.3.5 Explanation through graph eigenvalues

We now provide an intuitive explanation of the degradation of performance of the symmetric bidirectional architecture with increasing  $N$  when the loop transfer function has two integrators. This explanation uses the spectral properties of the interconnection graph. In particular, the minimum eigenvalue of the Dirichlet Laplacian of the interconnection graph is seen to have a profound impact on the performance loss. We will need the following result, which is also used in all the proofs of the theorems of this chapter.

**Lemma 7.3.1.** *consider the matrix  $L$ , as defined in (7.10). Let  $\lambda_{min}$  be the smallest eigenvalue of  $L$ , and let  $u_1 := [u_{11}, u_{21}, \dots, u_{N1}]^T$  be a unit-norm eigenvector of  $L$  corresponding to  $\lambda_{min}$ . Then the following are true:*

$$\begin{aligned} \frac{1}{N^2} < \lambda_{min} &\leq \frac{\pi^2}{N^2}, \quad \forall N. \\ |u_{11}| &> N^{-1/2}, \quad \forall N. \end{aligned}$$

Now the explanation. Since  $L$  is symmetric,  $\exists U \in \mathbb{R}^{N \times N}$  with  $U^T U = U U^T = I$  s.t.  $L = U \Lambda U^T$  where  $\Lambda$  is a real diagonal matrix containing the eigenvalues of  $L$  and  $U = [u_1, u_2, \dots, u_N]$ ,  $u_i$  being a unit-norm eigenvector of  $L$  corresponding to the  $i$ th eigenvalue. The eigenvalues are arranged as

$$\lambda_{min} \leq \lambda_2 \leq \dots \leq \lambda_{N-1} \leq \lambda_{max}.$$

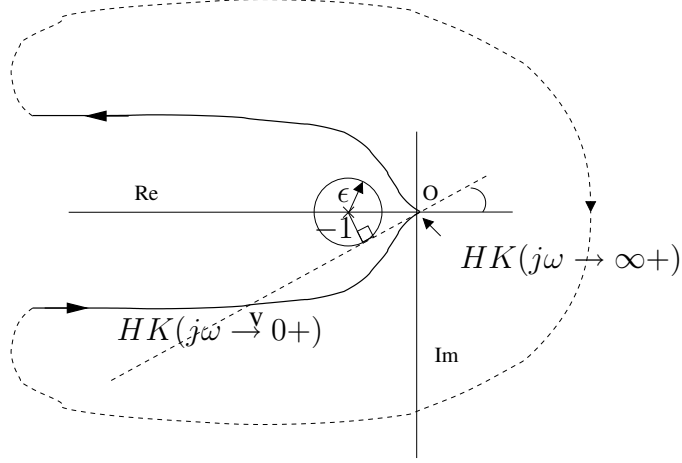


Figure 7.3. Nyquist plot of  $H(s)K(s)$  when  $H(s)K(s)$  has two integrators.

Hence,

$$\begin{aligned}
 I + H(s)K(s)L &= U(I + H(s)K(s)\Lambda)U^T \\
 \Rightarrow (I + H(s)K(s)L)^{-1} &= U(I + H(s)K(s)\Lambda)^{-1}U^T. \\
 \Rightarrow G_{x_{oe}}(s) &= U\Psi(s)U^T\phi_1
 \end{aligned}$$

where

$$\Psi(s) = \begin{bmatrix} \frac{1}{1+\lambda_{\min}H(s)K(s)} & & & \\ & \frac{1}{1+\lambda_2H(s)K(s)} & & \\ & & \ddots & \\ & & & \frac{1}{1+\lambda_{\max}H(s)K(s)} \end{bmatrix}$$

Therefore, for a fixed omega, the 2-norm of  $G_{de}(j\omega)$  is given by

$$\|G_{x_{oe}}(j\omega)\| = \max_i \left| \frac{1}{1 + \lambda_{\max}H(j\omega)K(j\omega)} \right| \quad (7.12)$$

When  $HK$  has two integrators, its magnitude at very low frequencies is arbitrary large and phase is arbitrarily close to 0. As shown schematically in Figure 7.3, the magnitude of the phasor  $|1 + \lambda_{\min}H(j\omega)K(j\omega)|$  can be made smaller

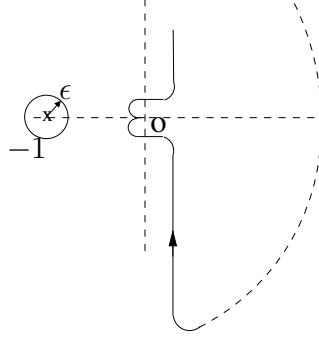


Figure 7.4. Nyquist plot of  $H(s)K(s)$  when  $H(s)K(s)$  has one integrator.

than an arbitrary  $\epsilon$  by choosing a small enough  $\omega$  and a correspondingly small  $\lambda_{\min}$ , which in turn can be done by choosing a large enough  $N$ , since  $\lambda_{\min} = \Theta(1/N^2)$ . This is the reason that when the number of vehicles increases without bound,  $\|G_{x_{oe}}\|_{\infty}$  grows without bound.

If  $H(s)K(s)$  has only one integrator, since  $H(s)$  has at least one integrator by assumption,  $K(s)$  has no integrator. It is clear from the Nyquist plot of  $H(s)K(s)$ , a sample one is shown in Figure 7.4 that the length of the phasor  $|1 + \lambda_i H(j\omega)K(j\omega)|$  can be kept larger than a positive constant for all  $i$ , no matter how large  $N$  is, by an appropriate choice of  $K$ . The reason for this is that maximum  $\lambda_i$  is 4 and the eigenvalues smaller than 1 only increase  $|1 + \lambda_i H(j\omega)K(j\omega)|$ . This is the reason that  $\|G_{x_{oe}}\|_{\infty}$  stays bounded no matter what  $N$  is.

## 7.4 Comments and open problems

Control of vehicular platoons, usually posed as an interconnection of point masses, has practical implications for automated highways. However, even after five decades of study, it has proved difficult to come up with a satisfactory solution, that consists of an interconnection architecture and an associated con-

trol algorithm that ensures guaranteed level of stability margin and robustness to disturbances.

The bidirectional architecture was the natural one to study when the limitations of the predecessor-following architecture became apparent. Although the motivation for a bidirectional architecture is clear, the reason for studying the *symmetric* version of it is little more than convenience. Since designing separate controllers for individual vehicles is a challenging task, the problem is simplified by imposing an arbitrary symmetry. The results in this chapter indicate that the symmetric bidirectional architecture suffers from limitations that cannot be ameliorated by better controller design.

The immediate question is, of course, if it might be possible to do better by removing the symmetry in the interconnection architecture. This question is answered in the positive in the next chapter.

The other issue identified here is that the difficulty of the platoon problem comes from the interplay of the interconnection topology (manifested in the graph eigenvalues) and the unbounded gain and large negative phase of the vehicle model at low frequencies. It seems that disturbance amplification occurs regardless of whether the vehicle model has one or two integrators, but the single integrator case is less sensitive to disturbances caused by the lead vehicle. An answer to the question of what happens when the vehicle model and the controller each has one integrator has not been answered here and needs to be examined.

In the work of Zhang *et al.* [153] on bidirectional control with non-linear plant models and non-linear controller, the proposed controller was able to achieve closed loop stability without any “slinky type” effects. However, due to the exceedingly complex vehicle model used (with engine speed, brake torque, manifold

pressure etc all appearing in the control design), it is not clear how the design was able to avoid slinky-type effects. Still, the results in [153] indicate that the perhaps the problem of vehicular platoon control should be studied in the non-linear setting for a better chance at avoiding the difficulties identified in this chapter and elsewhere.

## 7.5 Proofs

First we provide the proof of Lemma 7.3.1, which is used in all subsequent proofs.

*Proof of Lemma 7.3.1.* The matrix  $M$  is non-singular, since  $\det(M) = 1$ , from (7.5). Since  $L$  is a product of a square matrix  $M$  and its transpose, and  $M$  is non-singular,  $L$  is positive definite [100]. Since  $L = L^T$ , all eigenvalues of  $L^{-1}$  are positive real. So the smallest eigenvalue of  $L$  is the inverse of the largest eigenvalue of  $L^{-1}$ . Note that  $L^{-1}$  is given by

$$L^{-1} = \begin{bmatrix} N & N-1 & \dots & 2 & 1 \\ N-1 & N-1 & \dots & 2 & 1 \\ & & \ddots & & \\ 2 & 2 & \dots & 2 & 1 \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix}$$

To prove it, simply multiply the matrix with  $L$  and check that an identity matrix results. From Gerschgorin circle theory, we know that an upper bound for the largest eigenvalue of  $L^{-1}$  is  $\sum(1 + 2 + \dots + N) < N^2$ . Therefore, a lower bound for the smallest eigenvalue of  $L$  is  $1/N^2$ . That is,  $\lambda_{min} > 1/N^2$ . To get the upper bound on  $\lambda_{min}$ , let us write  $L$  as

$$L = \left[ \begin{array}{c|c} 1 & -\phi_{1(N-1)}^T \\ \hline \phi_{1(N-1)} & L_1 \end{array} \right]_{N \times N}$$

where  $\phi_{1(N-1)}$  is the first element of the canonical basis vector of  $\mathbb{R}^{N-1}$  and  $L_1 \in \mathbb{R}^{N-1 \times N-1}$ . It turns out that  $L_1$  is the so-called finite-difference matrix. From Cauchy's Interlacing Theorem, we know that  $\lambda_{\min} \leq \mu_{\min}$ , where  $\mu_{\min}$  is the smallest eigenvalue of  $L_1$ . It is known [100] that  $\mu_{\min} = 4 \sin^2(\pi/2N)$ . Moreover, for  $\theta > 0$ ,  $\sin \theta \leq \theta$ . Hence,  $\mu_{\min} \leq \pi^2/N^2$ , which establishes the upper bound on  $\lambda_{\min}$ .

To prove the second statement, note that since  $u_1$  is an eigenvector of  $L$  corresponding to the smallest eigenvalue of  $L$ ,  $u_1$  is also an eigenvector of  $L^{-1}$  corresponding to its largest eigenvalue. Since  $L^{-1}$  is a positive matrix, Perron-Frobenius theory tells us that  $|u_1| := \{|u_{11}|, \dots, |u_{N1}|\}$  is also an eigenvector of  $L^{-1}$  corresponding to its largest eigenvalue and that  $|u_1|$  is a positive vector. Thus, we can make the unit-norm eigenvector  $u_1$  of  $L$ , corresponding to  $\lambda_{\min}$ , consist entirely of positive numbers. Let's write down the equation  $Lu_1 = \lambda_{\min}u_1$  in expanded form:

$$\begin{bmatrix} u_{11} - u_{21} \\ -u_{11} + 2u_{21} - u_{31} \\ -u_{21} + 2u_{31} - u_{41} \\ \dots \end{bmatrix} = \begin{bmatrix} \lambda_{\min}u_{11} \\ \lambda_{\min}u_{21} \\ \lambda_{\min}u_{31} \\ \dots \end{bmatrix}$$

It is easy to check from these equations and the positivity of  $u_{i1}$ 's that  $u_{i1}$ 's form a decreasing sequence:  $u_{11} > u_{21} > \dots u_{N1} > 0$ . Since  $\sum u_{i1}^2 = 1$ , it follows that  $u_{11}^2 > 1/N$ . This proves the Lemma. ■

*Proof of Theorem 7.3.1.* We start by proving the second statement, which is similar to the results established by Fax *et. al.* [128] about the role played by the eigenvalues of the graph Laplacian on formation stability. It is also easy to see once we simplify equation (7.8). Since  $L$  is symmetric,  $\exists U \in \mathbb{R}^{N \times N}$  with  $U^T U = U U^T = I$  s.t.  $L = U \Lambda U^T$  where  $\Lambda$  is a real diagonal matrix containing the eigenvalues of  $L$

and  $U = [u_1, u_2, \dots, u_N]$ ,  $u_i$  being a unit-norm eigenvector of  $L$  corresponding to the  $i$ th eigenvalue. The eigenvalues are arranged as

$$\lambda_{min} \leq \lambda_2 \leq \dots \leq \lambda_{N-1} \leq \lambda_{max}.$$

Hence,

$$\begin{aligned} I + H(s)K(s)L &= U(I + H(s)K(s)\Lambda)U^T \\ \Rightarrow (I + H(s)K(s)L)^{-1} &= U(I + (HK)\Lambda)^{-1}U^T. \end{aligned}$$

Using the above and (7.8), we get

$$\begin{aligned} G_{x_o e}(s) &= U(I + H(s)K(s)\Lambda)^{-1}U^T\phi_1 \\ &= U\Psi(s)U^T\phi_1, \end{aligned} \tag{7.13}$$

where the matrix  $\Psi(s) \in \mathbb{C}^{N \times N}$  is defined as

$$\begin{aligned} \Psi(s) &:= (I + H(s)K(s)\Lambda)^{-1} \\ &= \begin{bmatrix} \frac{1}{1 + \lambda_{min}H(s)K(s)} & & \\ & \ddots & \\ & & \frac{1}{1 + \lambda_{max}H(s)K(s)} \end{bmatrix}. \end{aligned} \tag{7.14}$$

This gives us, with (7.13), that

$$G_{x_o e}(s) = \begin{bmatrix} \sum_{i=1}^N \frac{1}{1 + \lambda_i H(s)K(s)} u_{1i}^2 \\ \sum_{i=1}^N \frac{1}{1 + \lambda_i H(s)K(s)} u_{1i} u_{2i} \\ \dots \\ \sum_{i=1}^N \frac{1}{1 + \lambda_i H(s)K(s)} u_{1i} u_{Ni} \end{bmatrix} \tag{7.15}$$

It is clear now that for the closed loop to be stable, each of the transfer functions  $1/(1 + \lambda_i H(s)K(s))$  for  $i \in \{1, 2, \dots, N\}$  must be stable. As a consequence, there

cannot be any unstable pole zero cancellation between  $H(s)$  and  $K(s)$ . Since  $H(s)$  has at least one integrator by assumption,  $K(s)$  cannot have any zeros at 0.

To prove (1), we consider the root locus of the system  $1 + \lambda_{min}H(s)K(s)$ . Suppose  $H(s)K(s)$  has three integrators. Then, at least one of the branches of the root loci will depart to the right half plane for an arbitrarily small value of  $\lambda_{min}$ , even though it may eventually return to the left half plane for a large enough value. Since  $\lambda_{min}$  can be arbitrarily small for  $N$  arbitrarily large (Lemma 7.3.1), this means that  $1/(1 + \lambda_{min}H(s)K(s))$  will be unstable for a large enough  $N$ . Thus,  $H(s)K(s)$  cannot have three integrators. The extension of these arguments to the case of more than three integrators is trivial.

To prove (3), let  $C(s) = N_c(s)/D_c(s)$  where  $N_c(s)$  and  $D_c(s)$  are coprime polynomials. From the above,  $C(s)$  cannot have zeros at the origin. Therefore  $C(s)$  does not have poles or zeros at the origin, so  $N_c(0) \neq 0$  and  $D_c(0) \neq 0$ . Consider the case when  $H(s)K(s)$  has two integrators, so the characteristic polynomial of  $1/(1 + \lambda_{min}H(s)K(s))$  is  $s^2D_c(s) + \lambda_{min}N_c(s)$ . If  $N_c(0) < 0$ , then  $\lambda_{min}N_c(0) < 0$  and the closed loop will have at least one unstable pole. Thus  $N_c(0) > 0$ . The coefficient of  $s^2$  in the characteristic polynomial is  $D_c(0) + \lambda_{min}c_2$ , where  $c_2$  is the coefficient of  $s^2$  in  $N_c(s)$ . If  $D_c(0) < 0$  the coefficient of  $s^2$  will be negative when  $\lambda_{min}$  is small enough, i.e., for a large enough  $N$ , even when  $c_2$  is positive. This will make the closed loop unstable. Thus, in order to have closed loop stability for arbitrary  $N$ , we must have  $D_c(0) > 0$ . Hence,  $C(0) > 0$ . These arguments can be repeated for the case when  $H(s)K(s)$  has one integrator, and we arrive at the same result. This proves the theorem. ■

*Proof of Theorem 7.3.2.* When  $H(s)K(s)$  has a double integrator, we can represent  $H(s)K(s)$  as  $C(s)/s^2$ , where  $C(s)$  does not have poles or zeros at zero and

$C(0) > 0$ . This follows from theorem 7.3.1. Consider the spacing error of the  $k$ th vehicle. Since  $x_o(t) = v_o t$ , so  $X_o(s) = v_o/s^2$ . This, together with equations (7.15) and (7.7) gives us

$$sE_k(s) = \sum_{i=1}^N \frac{sv_o}{s^2 + \lambda_i C(s)} u_{1i} u_{ki}$$

Since  $K(s)$  stabilizes the platoon dynamics,  $1/(s^2 + \lambda_i C(s))$  is a stable transfer function for  $i \in \{1, 2, \dots, N\}$  and therefore  $\lim_{s \rightarrow 0} sv_o/(s^2 + \lambda_i C(s)) = 0$ . Since  $u_{ij}$ 's are bounded numbers,  $sE_k(s) \rightarrow 0$  as  $s \rightarrow 0$ . Hence, from the Final Value Theorem,  $\lim_{t \rightarrow \infty} \bar{e}(t) = \lim_{s \rightarrow 0} s\bar{E}(s) = 0$ . This proves the first statement of the theorem.

Now we consider the case of  $H(s)K(s)$  having only one integrator. We can represent  $H(s)K(s)$  as  $C(s)/s$  where  $C(s)$  doesn't have poles or zeros at the origin and  $C(0) > 0$  (theorem 7.3.1). Since  $X_o(s) = v_o/s^2$ , we have

$$s\bar{E}(s) = U s \Psi(s) \frac{v_o}{s^2} U^T \phi_1 = U Q(s) U^T \phi_1,$$

where  $Q(s)$  is defined as

$$Q(s) := \begin{bmatrix} \frac{v_o}{s + \lambda_{min} C(s)} & & \\ & \ddots & \\ & & \frac{v_o}{s + \lambda_{max} C(s)} \end{bmatrix}.$$

Hence, once again from the Final Value Theorem,

$$\lim_{t \rightarrow \infty} \bar{e}(t) = \lim_{s \rightarrow 0} s\bar{E}(s) = U Q(0) U^T \phi_1 := e_\infty,$$

which is a constant vector. Thus the steady state error converges to a constant vector.

To prove that  $e_\infty$  grows unbounded with  $N$ , note that

$$\|\bar{e}_\infty\|_2^2 = \phi_1^T U Q(0)^T Q(0) U^T \phi_1. \quad (7.16)$$

Since  $U^T \phi_1$  is the first row of  $U$  and  $Q(0)$  is a real diagonal matrix, we can reduce (7.16) to

$$\|\bar{e}_\infty\|_2 = \left( \sum_{i=1}^N \left( \frac{v_o}{\lambda_i C(0)} \right)^2 u_{1i}^2 \right)^{1/2} > \frac{v_o}{\lambda_{\min} C(0)} |u_{11}|.$$

From lemma 7.3.1, we have  $1/\lambda_{\min} > N^2/\pi^2$  and  $|u_{11}| > 1/N^{1/2}$ . Using these in the above, we get  $\|\bar{e}_\infty\|_2 > \gamma N^{3/2}$ , where  $\gamma = v_o/C(0)\pi^2$ . Since this lower bound is an increasing function of  $N$ , the second part of the theorem follows immediately. ■

The following technical result will be needed for the proof of Theorem 7.3.3.

**Lemma 7.5.1.** *Let  $C(s)$  be a SISO transfer function that has no poles or zeros at the origin and  $C(0) > 0$ . Then,  $\exists \beta \in (0, +\infty)$  and  $\exists N_o \in \mathbb{N}$  such that  $\forall N > N_o$ ,*

$$\sup_{\omega} \left| \frac{1}{1 - \frac{\lambda_{\min}(N)C(j\omega)}{\omega^2}} \right| > \beta N.$$

where  $\lambda_{\min}(N)$  is the smallest eigenvalue of the matrix  $L \in \mathbb{R}^{N \times N}$  defined in (7.10). □

*Proof of Lemma 7.5.1.* First we will establish that  $|C(j\omega) - C(0)| < \omega\gamma$  for some positive constant  $\gamma$  when  $\omega$  is small enough. Let  $C(s) = N_c(s)/D_c(s)$ , where  $N_c(s)$  and  $D_c(s)$  are coprime polynomials in  $s$  (with real coefficients) with degrees  $m$  and  $n$ , respectively. We write down

$$C(s) = \frac{N_c(s)}{D_c(s)} = \frac{z_m s^m + \cdots + z_1 s + z_o}{p_n s^n + \cdots + p_1 s + p_o}$$

where  $z_o$  and  $p_o$  are non-zero since  $C(s)$  does not have poles or zeros at the origin. Expanding the expression for  $C(s) - C(0)$  and doing a little algebra, we see that

$$C(s) - C(0) = \frac{s^k Q(s)}{D(s)p_o},$$

where  $Q(s)$  is a polynomial in  $s$  with a non-zero constant term and  $k \geq 1$ . Since  $Q(s)$  and  $D(s)$  both have non-zero constant terms,  $\lim_{\omega \rightarrow 0} \frac{Q(j\omega)}{D(j\omega)p_o} = \frac{Q(0)}{D(0)p_o} \neq 0$ . Therefore,  $\exists \omega_o$  s.t. if  $|\omega| < \omega_o$ , then  $|\frac{Q(j\omega)}{D(j\omega)p_o}| < |\frac{Q(0)}{D(0)p_o}| + 1 =: \gamma$ . Therefore we get that there exist  $\omega_o > 0, \gamma > 0$  and an integer  $k \geq 1$  s.t.,  $\forall |\omega| < \min(1, \omega_o)$ ,

$$|C(j\omega) - C(0)| \leq |\omega|^k \gamma \leq |\omega| \gamma. \quad (7.17)$$

Define

$$f(\omega) = \left| \frac{1}{1 - \frac{\lambda_{\min}(N)C(j\omega)}{\omega^2}} \right|.$$

Pick  $N_o$  such that  $\omega^* := \sqrt{\lambda_{\min}(N)C(0)} \in (0, \min(1, \omega_o))$ ,  $\forall N > N_o$ . Hence,

$$f(\omega^*) = \frac{1}{|1 - \frac{C(j\omega^*)}{C(0)}|} = \frac{C(0)}{|C(j\omega^*) - C(0)|} > C(0)/\gamma\omega^*.$$

The last inequality follows from (7.17). Substituting the value of  $\omega^*$ , we get

$$f(\omega^*) > \frac{C(0)^{1/2}}{\gamma\lambda_{\min}^{1/2}}, \quad \forall N > N_o. \quad (7.18)$$

From lemma 7.3.1, we know that  $1/\lambda_{\min}(N) \geq N^2/\pi^2$ . Using this in the inequality (7.18), we get

$$f(\omega^*) > \beta N, \quad \forall N > N_o \quad (7.19)$$

where  $\beta := (C(0)/\gamma^2\pi^2)^{1/2}$  is a positive constant. This proves the lemma. ■

Now we are ready to prove Theorem 7.3.3.

*Proof of Theorem 7.3.3.* Since  $H(s)K(s)$  has two integrators,  $H(s)K(s)$  can be written as  $C(s)/s^2$ . From theorem 7.3.1, it follows that  $C(s)$  cannot have poles or zeros at 0 and  $C(0) > 0$ . From (7.13), we get

$$\|G_{xoe}(s)\|_2 = \sqrt{G_{xoe}^* G_{xoe}} = \sqrt{\phi_1^T U \Psi^*(s) \Psi(s) U^T \phi_1}.$$

Since the vector  $U^T \phi_1$  is the first row of  $U$ , using (7.14), this reduces to

$$\|G_{x_{oe}}(s)\|_2 = \left( \sum_{i=1}^N u_{1i}^2 \left| \frac{1}{1 + \lambda_i H(s)K(s)} \right|^2 \right)^{1/2} \quad (7.20)$$

The  $\mathcal{H}_\infty$  norm of the transfer function vector  $G_{x_{oe}}$  is:

$$\|G_{x_{oe}}\|_\infty = \sup_{\omega} \|G_{x_{oe}}(j\omega)\|_2. \quad (7.21)$$

Thus,

$$\|G_{x_{oe}}\|_\infty > \sup_{\omega} \left( \left| \frac{1}{1 + \lambda_{\min} H(j\omega)K(j\omega)} \right| |u_{11}| \right), \quad (7.22)$$

We can now apply the result established in lemma 7.5.1 to claim that  $\exists \beta \in (0, +\infty)$  and  $\exists N_o \in \mathbb{N}$  such that

$$\sup_{\omega} \left| \frac{1}{1 + \lambda_{\min}(N) H(j\omega)K(j\omega)} \right| > \beta N \quad \forall N > N_o.$$

From lemma 7.3.1, we know that  $|u_{11}| > 1/\sqrt{N}$ . Using these two inequalities in (7.22), we get

$$\|G_{x_{oe}}\|_\infty > \beta N^{1/2} \quad \forall N > N_o.$$

Since this lower bound grows unbounded as  $N$  increases, the result follows immediately. ■

*Proof of Theorem 7.3.5.* When  $K(s)$  has no integrators, and  $HK$  has  $k$  integrators, where  $k$  is either one or two,  $H$  must be of the form  $H(s) = H_1(s)/s^k$  with  $H_1(0) > 0$ . From (7.9) and using  $L = M^T M$ , we get

$$G_{de}(s) = H(s)[I + H(s)K(s)M^T M]^{-1} M^T = H(s)s^k [M^{-T} s^k + C(s)M]^{-1} \quad (7.23)$$

Therefore,

$$\|G_{de}(0)\| = \frac{H_1(0)}{C(0)} M^{-1} \Rightarrow \|G_{de}(0)\| = \frac{H_1(0)}{C(0)} N,$$

since

$$M^{-1} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & 1 & \dots & 1 \\ \vdots & \dots & \ddots & \vdots \\ 0 & \dots & 0 & 1 \end{bmatrix} \in \mathbb{R}^{N \times N},$$

whose 2-norm is  $N$ . This implies  $\|G_{de}\|_{\infty} = \sup_{\omega} \|G_{de}(j\omega)\| \geq \|G_{de}(0)\| \geq cN$  for some constant  $c$  independent of  $N$ . This proves the statement. ■

# Chapter 8

## Control of vehicular platoons: asymmetric bidirectional control

### 8.1 Introduction

In this chapter we revisit the problem examined in the previous chapter - that of decentralized control of a string of vehicles moving in a straight line in order to maintain constant inter-vehicular separation. In the general bidirectional case, there is no reason for a vehicle's controller to put equal weights on the front spacing error (i.e., the error w.r.t. to the preceding vehicle's relative position) and back spacing error (i.e., the error w.r.t. to the following vehicle's relative position). Moreover, the controllers in one vehicle should be allowed to be different from those in other vehicles. The difficulty of this general problem is, however, that we are faced with the task of designing  $2N$  controllers when there are  $N$  vehicles.

Due to the challenging nature of this problem, we resort to a continuum approximation of the platoon dynamics in the form of a partial differential equa-

tion(PDE). The continuum approximation turns out to be quite useful in providing insight into the problem, which led to the improved, “mistuning”-based, design described in this chapter. Although the continuum approximation is made under the assumption of large number of vehicles, the resulting design and analysis show that the benefits are tangible even for small number of vehicles.

*Chapter organization:* We start with a summary of the results in Section 8.2 and state the problem in Section 8.3 in formal terms. Unlike some of the previous chapters, we do not have a “results” section, since we need to describe the PDE model of the platoon dynamics in order to state the results. Section 8.4 describes the derivation of the PDE model. In section 8.5 the PDE is analyzed to explain the loss of stability with increasing number of vehicles, and section 8.6 describes how to ameliorate such loss of stability by mistuning. Section 8.7 reports time-domain simulation results that show the benefit of mistuning.

## 8.2 Contributions and prior work

The contributions of the work reported in this chapter are briefly summarized below.

1. In order to facilitate the analysis, we derive a linear partial differential equation (PDE) based continuous analogue of the (spatially) discrete platoon dynamics. The PDE model is inspired by the extensive literature on PDE based models of traffic dynamic (see the review [162] and references therein, and the PDE model of a string of vehicles considered in [163]). However, a PDE based model of a controller platoon is a novel contribution of our work.

2. The PDE model is used to derive a controller independent conclusion on stability with symmetric bi-directional architecture. In particular, the behavior of the least stable eigenvalue of the discrete platoon dynamics is predicted by analyzing the PDE spectra. We show that the least stable closed-loop eigenvalue approaches zero as  $O(\frac{1}{N^2})$ . This prediction is confirmed by numerical computation.
3. We show that an arbitrary small perturbation (asymmetry) in the controller gains from their nominal (symmetric) values can improve the closed-loop damping such that the least stable eigenvalue now approaches 0 only as  $O(\frac{1}{N})$ . Numerical computations of eigenvalues in discrete platoons is used to validate these results.

Perhaps more important than the improvement itself is the intuition into the problem that the PDE provides, which made the improvement possible in the first place. The PDE reveals, better than the discrete state-space equation does, the underlying cause of progressive loss of stability with a symmetric bidirectional architecture and suggests a mistuning-based approach to improve the stability margin by introducing asymmetry. In particular, *forward-backward asymmetry* in the control is seen to be beneficial. The asymmetry refers to the assignment of controller gains such that a vehicle utilizes information from the preceding and following vehicles differently. We also show how to achieve the best improvement in closed-loop stability by exploiting this asymmetry.

*Prior work:* Prior work on the control of vehicular platoons has been reviewed in the previous chapter; see Section 7.2.1. The idea of using non-identical controllers to improve robustness to disturbance of the closed loop platoon has

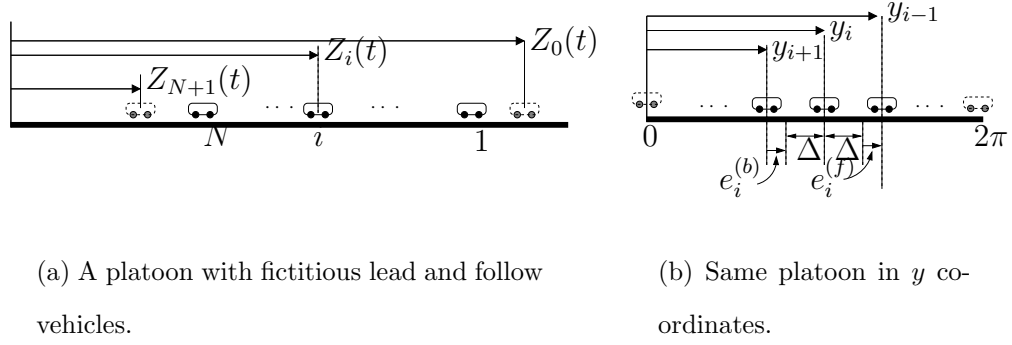


Figure 8.1. A platoon with  $N$  vehicles moving in one dimension.

been considered in [164]. However, in the design proposed in [164], the controller gains at individual vehicles grow without bound as  $N$  increases. In contrast, the mistuning based design proposed in this chapter keeps controller gains uniformly bounded within any prescribed value, independent of the number of vehicles. The role of asymmetry in the interconnection architecture on improving closed loop stability is novel.

PDE modeling of traffic flow is quite well developed, see [162] and references therein for a thorough review of this topic. We note that the mistuning based approaches have been used for stability augmentation in many structural applications; see [165–168] for some recent references.

### 8.3 Problem statement

Consider a platoon of  $N$  identical vehicles moving in a straight line as shown schematically in Figure 8.1(a). For these vehicles, we consider the following two scenarios as tabulated in Table 8.1. In scenario I, we introduce (after [158, 169]) a fictitious lead vehicle and a fictitious follow vehicle, indexed as 0 and  $N + 1$  respectively. Their behavior is specified by imposing a constant velocity trajec-

ries as  $Z_0(t) = V_d t$  and  $Z_{N+1} = V_d t - (N + 1)\Delta$ . In scenario II, only a fictitious lead vehicle with index  $i = 0$  with  $Z_0(t) = V_d t$  is introduced. For the last vehicle in the platoon in scenario II, there is no follower vehicle and it uses information only from its predecessor to maintain a constant gap.

For the ease of analysis and design, we take the following simplifying assumptions:

1. Every vehicle is a fully actuated point mass without damping (i.e., a double integrator)
2. Every vehicle employs a static gain feedback control law.

These simplifications are done for ease of analysis only; the results are seen to be valid more generally. Let  $Z_i(t)$  and  $V_i(t)$  denote the position and the velocity, respectively, of the  $i^{\text{th}}$  vehicle for  $i = 1, 2, \dots, N$ . Since the inter-connected platoon dynamics are of primary interest, a simple double integrator is used to model the essential dynamics of an individual vehicle:

$$\ddot{Z}_i = U_i,$$

where  $U_i$  is the control (engine torque) applied on the  $i^{\text{th}}$  vehicle. Formally, such a model arises after the velocity dependent drag and other non-linear terms have been eliminated by using feedback linearization [145, 146]. The control objective is to maintain a constant inter-vehicular distance  $\Delta$  and a constant velocity  $V_d$  for every vehicle.

To facilitate the analysis, consider a co-ordinate change

$$y_i = 2\pi\left(\frac{Z_i(t) - V_d t + L}{L}\right), \quad v_i = 2\pi\frac{V_i - V_d}{L}, \quad (8.1)$$

Scenario	Length $L$	Leader	Follower
I	$(N + 1)\Delta$	$\tilde{v}_0 = 0$	$\tilde{v}_{N+1} = 0$
II	$N\Delta$	$\tilde{v}_0 = 0$	–

Table 8.1. The two scenarios – one with a fictitious lead vehicle, the other with fictitious lead and follow vehicles.

where  $L$  denotes the platoon length, which equals  $(N + 1)\Delta$  in scenario I and  $N\Delta$  in scenario II. Figure 8.1(b) depicts the schematic of the platoon in the new co-ordinates. The normalization ensures that  $y_0(t) \equiv 2\pi$ ,  $y_i(t) \in [0, 2\pi]$ , and  $y_{N+1}(t) \equiv 0$  ( $y_N(t) = 0$ ) in scenario I (II). Here, we have implicitly assumed that deviations of the vehicle positions and velocities from their desired values are small.

In the normalized co-ordinate, the dynamics of the  $i^{\text{th}}$  vehicle is described by

$$\ddot{y}_i = u_i,$$

where  $u_i := 2\pi U_i/L$ . The desired spacing and velocities are

$$\delta := \frac{\Delta}{L/2\pi}, \quad v_d := \frac{V_d - V_d}{L/2\pi} = 0,$$

and the desired position of the  $i^{\text{th}}$  vehicle is

$$y_i^d(t) \equiv 2\pi - i\delta. \tag{8.2}$$

The position and velocity errors for the  $i^{\text{th}}$  vehicle in the  $y$  co-ordinate are:

$$\tilde{y}_i(t) = y_i(t) - y_i^d(t), \quad \tilde{v}_i = v_i - v_d = v_i.$$

We note that  $\tilde{v}_0 = \tilde{v}_{N+1} = 0$  for the fictitious lead and follow vehicles.

For the purposes of control, it is useful to introduce the front and back relative (position) errors for the  $i^{\text{th}}$  vehicle:

$$\begin{aligned} e_i^{(f)} &= \frac{Z_{i-1} - Z_i - \Delta}{L/2\pi} = y_{i-1} - y_i - \delta, \\ e_i^{(b)} &= \frac{Z_i - Z_{i+1} - \Delta}{L/2\pi} = y_i - y_{i+1} - \delta, \quad \text{for } i = \{1, \dots, N\}. \end{aligned}$$

The quantity  $e_i^{(f)}$  denotes the *front* relative position error between the  $i^{\text{th}}$  and its predecessor  $(i - 1)$  vehicle, and  $e_i^{(b)}$  denotes the *back* relative position error between the  $i^{\text{th}}$  and its follower  $(i + 1)$  vehicle. The relative errors, including the velocity error, can be obtained in practice by on-board devices such as radars, GPS and speed sensors. Consistent with the decentralized bidirectional linear control architecture, the control  $u_i$  for the  $i^{\text{th}}$  vehicle is assumed to depend only on 1) its velocity  $\tilde{v}_i$ , and 2) the relative position errors between itself and its immediate neighbors. That is,

$$u_i = k_i^{(f)} e_i^{(f)} - k_i^{(b)} e_i^{(b)} - b_i \tilde{v}_i, \quad (8.3)$$

where  $k_i^{(\cdot)}, b_i$  are positive constants. The first two terms are used to compensate for any deviation away from nominal with the predecessor and the follower vehicles respectively. The third term is used to obtain a zero steady-state error in velocity. In principle, relative velocity errors between neighboring vehicles can also be incorporated into the control, but we do not examine this situation here.

Equation (8.3) represents control using state feedback, albeit only with local (nearest neighbor) information. Analysis of this controller structure is relevant even if there are additional dynamic elements in the controller. First, a dynamic controller cannot be allowed to have a zero at the origin. The reason is that for a constant velocity reference, such a pole-zero cancellation will lead to steady-state errors that grow without bound as  $N$  increases (see Theorem 7.3.2 in the previous

chapter). Second, a dynamic controller cannot have an integrator either. For if it does, the closed-loop platoon dynamics become unstable for a sufficiently large value of  $N$  (see Theorem 7.3.1). As a result, any *allowable* dynamic compensator must essentially act as a static gain at low frequencies. Furthermore, the results of [160] indicate that the principal challenge in controlling large platoons arises from the double integrator with its unbounded gain and large negative phase at low frequencies (see Section 7.3.5). Hence, the limitation and its amelioration discussed here only with the local state feedback of (8.3) is also relevant to the case where additional dynamic elements appear in the control.

To describe the closed-loop dynamics of the platoon, define

$$\begin{aligned}\tilde{\mathbf{y}} &:= [\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_N]^T, \\ \tilde{\mathbf{v}} &:= [\tilde{v}_1, \dots, \tilde{v}_N]^T.\end{aligned}$$

For scenario I with fictitious lead and follow vehicles, the control law (8.3) yields the following closed loop dynamics.

$$\begin{bmatrix} \dot{\tilde{\mathbf{y}}} \\ \dot{\tilde{\mathbf{v}}} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & I \\ -K_I^{(f)} M^T - K_I^{(b)} M & -B \end{bmatrix}}_{A_{L-F}} \begin{bmatrix} \tilde{\mathbf{y}} \\ \tilde{\mathbf{v}} \end{bmatrix} \quad (8.4)$$

where  $K_I^{(f)} = \text{diag}(k_1^{(f)}, k_2^{(f)}, \dots, k_N^{(f)})$ ,  $K_I^{(b)} = \text{diag}(k_1^{(b)}, k_2^{(b)}, \dots, k_N^{(b)})$ ,  $B = \text{diag}(b_1, b_2, \dots, b_N)$  and

$$M = \begin{bmatrix} 1 & -1 & 0 & \dots \\ 0 & 1 & -1 & \dots \\ \vdots & & \ddots & \ddots \\ & & & 1 & -1 \\ & & & \dots & 0 & 1 \end{bmatrix}.$$

For scenario II with a fictitious lead vehicle and no follow vehicle, the closed loop

dynamics are

$$\begin{bmatrix} \dot{\tilde{\mathbf{y}}} \\ \dot{\tilde{\mathbf{v}}} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & I \\ -K_{\text{II}}^{(f)} M^T - K_{\text{II}}^{(b)} M_o & -B \end{bmatrix}}_{A_L} \begin{bmatrix} \tilde{\mathbf{y}} \\ \tilde{\mathbf{v}} \end{bmatrix} \quad (8.5)$$

where  $K_{\text{II}}^{(f)} = K_{\text{I}}^{(f)}$ ,  $K_{\text{II}}^{(b)} = \text{diag}(k_1^{(b)}, k_2^{(b)}, \dots, k_{N-1}^{(b)}, 0)$ , and

$$M_o = \begin{bmatrix} 1 & -1 & 0 & \dots \\ 0 & 1 & -1 & \dots \\ \vdots & & \ddots & 0 \\ \dots & 0 & 0 & 0 \end{bmatrix}.$$

Our goal is to understand the progressive loss of closed loop stability with increasing  $N$  and to devise ways to ameliorate such a loss by appropriately choosing the controller gains. While in principle this can be done by analyzing the eigenvalues of the matrix  $A_{L-F}$  (scenario I) and of  $A_L$  (scenario II), we take an alternate route. When the number of vehicles  $N$  is large, we approximate the dynamics of the discrete platoon by a partial differential equation (PDE) which is used for analysis and control design.

## 8.4 Continuous model of an automated platoon

In this section, we develop a continuous PDE approximation of the (spatially) discrete platoon dynamics with bidirectional control. The PDE is derived with respect to a normalized spatial co-ordinate  $x \in [0, 2\pi]$ . We recall that the normalized location of the  $i^{\text{th}}$  vehicle (denoted as  $y_i$ ) too was defined with respect to this co-ordinate system. In effect, the two symbols  $x$  and  $y$  correspond to the same co-ordinate representation but are used here to distinguish the continuous and discrete formulations.

With respect to the normalized co-ordinate, every car is nominally assumed to

lie within an interval of length  $\delta$  (see Fig. 8.1(b)). For the purposes of continuous approximation, we smear each vehicle over its interval to get a constant mean density

$$\rho_0 \approx \frac{N}{2\pi} = \frac{1}{\delta} \quad (8.6)$$

for  $N$  vehicles in the platoon. Dynamics of the individual vehicles in the platoon create perturbations in the density, with the local density  $\rho(x, t)$  increasing (decreasing) as the cars move closer (apart). The starting point of macroscopic continuous models of traffic flow thus is the continuity equation, which relates the density  $\rho(x, t)$  (vehicles per unit characteristic length) at spatial co-ordinate  $x \in [0, 2\pi]$  and time  $t \in [0, \infty)$  with the velocity  $v(x, t)$ :

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} = 0.$$

In order to analyze small perturbations about the mean, we define the perturbed quantities  $\tilde{\rho}, \tilde{v}$  by the relations

$$\rho(x, t) = \rho_0 + \tilde{\rho}(x, t), \quad v(x, t) = 0 + \tilde{v}(x, t),$$

where the mean velocity is zero because of our choice of the co-ordinate system (see (8.1)). Even though  $v$  and  $\tilde{v}$  are the same, we use  $\tilde{v}$  to draw attention to the fact that the velocity is a small perturbation of the mean value. For such perturbations, the linearized continuity equation is given by

$$\frac{\partial \tilde{\rho}}{\partial t} + \rho_0 \frac{\partial \tilde{v}}{\partial x} = 0 \quad \Rightarrow \quad \frac{\partial \tilde{v}}{\partial x} = -\frac{1}{\rho_0} \frac{\partial \tilde{\rho}}{\partial t} \quad (8.7)$$

This equation is consistent with the physical intuition whereby a positive gradient in velocity (due to say the predecessor speeding up or the follower slowing down) will cause the local density to decrease. In order to study density perturbations,

one thus needs to specify the velocity which here arises due to the linearized momentum balance:

$$\frac{\partial v}{\partial t} = F(x, t) \Rightarrow \frac{\partial \tilde{v}}{\partial t} = u(x, t), \quad (8.8)$$

where  $F(x, t)$  is the acceleration due to control  $u(x, t)$  and possibly disturbance. Here, we focus only on the control. Since we are in the moving coordinate frame, the momentum equation has the partial derivative rather than the usual total derivative on the left hand side. Using (8.3), the control for the  $i^{\text{th}}$  vehicle in the platoon is of the form:

$$u_i(t) = \underbrace{u_i^{(pf)}(t) - u_i^{(pb)}(t)}_{u_i^{(p)}(t)} + u_i^{(v)}(t),$$

where

$$u_i^{(pf)}(t) := k_i^{(f)}(y_{i-1}(t) - y_i(t) - \delta), \quad u_i^{(pb)}(t) := k_i^{(b)}(y_i(t) - y_{i+1}(t) - \delta),$$

are the position dependent front and back control terms, and,

$$u_i^{(v)}(t) := -b\tilde{v}_i(t). \quad (8.9)$$

Corresponding to this discrete control law, we derive a continuous approximation

$$u(x, t) = \underbrace{u^{(pf)}(x, t) - u^{(pb)}(x, t)}_{u^{(p)}(x, t)} + u^{(v)}(x, t), \quad (8.10)$$

such that  $u^{(pf)}(y_i, t) = u_i^{(pf)}(t)$ ,  $u^{(pb)}(y_i, t) = u_i^{(pb)}(t)$ , and  $u^{(v)}(y_i, t) = u_i^{(v)}(t)$ . Now,

$$\begin{aligned} u_i^{(pf)} &= k_i^{(f)}(y_{i-1} - y_i - \delta) = k_i^{(f)}\left(1 - \frac{\delta}{y_{i-1} - y_i}\right)(y_{i-1} - y_i) \\ &\approx \int_{y_i}^{y_{i-1}} k_{pf}(x)\left(1 - \frac{\rho(x)}{\rho_0}\right)dx, \end{aligned}$$

where the approximation is obtained by smearing the control action over the interval  $[y_i, y_{i-1}]$  and substituting  $k_f(x)$  for the discrete control gain  $k_i^{(f)}$ . Since

$\rho = \rho_0 + \tilde{\rho}$ , we have

$$u_i^{(pf)}(t) \approx -\frac{1}{\rho_0} \int_{y_i}^{y_{i-1}} k_f(x) \tilde{\rho}(x, t) dx = -\frac{1}{\rho_0} [k_f \tilde{\rho}](x^+, t) \delta,$$

by the Mean Value Theorem, where  $x^+ \in [y_i, y_{i-1}]$ . Since  $u_i^{(pf)}(t) = u^{(pf)}(y_i, t)$  and  $\delta = 1/\rho_0$  (see (8.6)), we take

$$u^{(pf)}(x, t) = -\frac{1}{\rho_0^2} [k_f \tilde{\rho}](x^+, t), \quad \text{and} \quad u^{(pb)}(x, t) = -\frac{1}{\rho_0^2} [k_b \tilde{\rho}](x^-, t),$$

where  $x^+ \in [y_i, y_{i-1}]$  and  $x^- \in [y_{i+1}, y_i]$ . Using (8.10), we have

$$\begin{aligned} u^{(p)}(x, t) &= u^{(pf)}(x, t) - u^{(pb)}(x, t) \\ &= -\frac{1}{\rho_0^2} ([k_f \tilde{\rho}](x^+, t) - [k_b \tilde{\rho}](x^-, t)). \end{aligned}$$

In order to specify the control, one thus needs to approximate the terms on the right hand side as functions of  $(x, t)$ . For a small perturbation about a nominally symmetric bi-directional architecture, a valid approximation is obtained by taking  $x^+ - x^- \approx \delta$  which yields

$$\begin{aligned} u^{(p)}(x, t) &\approx -\frac{1}{\rho_0^2} \left[ [k_p^{(-)} \tilde{\rho}](x, t) + \frac{\delta}{2} \frac{\partial [k^{(+)} \tilde{\rho}]}{\partial x}(x, t) \right] \\ &= -\frac{1}{\rho_0^2} k^{(-)} \tilde{\rho} - \frac{1}{2\rho_0^3} \frac{\partial}{\partial x} (k^{(+)} \tilde{\rho}), \end{aligned} \tag{8.11}$$

where

$$k^{(+)} := k_f(x) + k_b(x), \quad k^{(-)} := k_f(x) - k_b(x). \tag{8.12}$$

The velocity feedback term in (8.9) has a continuous counterpart

$$u^{(v)}(x, t) = -b(x) \tilde{v}(x, t). \tag{8.13}$$

With the feedback control  $u(x, t) = u^{(p)}(x, t) + u^{(v)}(x, t)$ , where  $u^{(p)}(x, t)$  and  $u^{(v)}(x, t)$  are given by (8.11) and (8.13), the linearized momentum equation (8.8) becomes

$$\frac{\partial \tilde{v}}{\partial t} = - \left[ \frac{1}{\rho_0^2} k^{(-)} \tilde{\rho} + \frac{1}{2\rho_0^3} \frac{\partial}{\partial x} (\tilde{\rho} k^{(+)} ) + b \tilde{v} \right].$$

Upon differentiating both sides with respect to  $t$  and using the continuity equation (8.7) we obtain the PDE that describes small velocity perturbations  $\tilde{v}$  due to the inter-connected platoon dynamics:

$$\left( \frac{\partial^2}{\partial t^2} + b \frac{\partial}{\partial t} \right) \tilde{v} = \frac{1}{\rho_0} (k_p^{(-)} \frac{\partial \tilde{v}}{\partial x}) + \frac{1}{2\rho_0^2} \frac{\partial}{\partial x} (k_p^{(+)} \frac{\partial \tilde{v}}{\partial x}) \quad (8.14)$$

The boundary conditions for the PDE depend upon the dynamics of the first and the last vehicles in the platoon. For scenario I with a constant velocity fictitious and lead vehicles, the appropriate boundary conditions are of the Dirichlet type on both ends:

$$\tilde{v}(0, t) = \tilde{v}(2\pi, t) = 0. \quad \forall t \in [0, \infty) \quad (8.15)$$

For scenario II with the only a fictitious lead vehicle, the appropriate boundary conditions are of Neumann-Dirichlet type:

$$\frac{\partial \tilde{v}}{\partial x}(0, t) = \tilde{v}(2\pi, t) = 0. \quad \forall t \in [0, \infty) \quad (8.16)$$

### 8.4.1 Eigenvalue comparison

For preliminary validation purposes, we consider the simplest case where the position control gains are constant for every vehicle, i.e.,  $k_f(x) = k_b(x) = k_0$  and  $b(x) = b_0$ . In such a case  $k^{(-)}(x) \equiv 0$ ,  $k^{(+)}(x) \equiv 2k_0$  and the governing PDE (8.14) simplifies to

$$\left( \frac{\partial^2}{\partial t^2} + b_0 \frac{\partial}{\partial t} - \frac{k_0}{\rho_0^2} \frac{\partial^2}{\partial x^2} \right) \tilde{v} = 0 \quad (8.17)$$

Note that this is a damped wave equation with a wave speed of  $\frac{\sqrt{k_0}}{\rho_0}$ . The wave equation is consistent with the physical intuition that a symmetric bidirectional control architecture causes a disturbance to propagated equally in both directions.

Figure 8.2 compares the closed loop eigenvalues of a discrete platoon with  $N = 25$  vehicles and the PDE (8.17). The eigenvalues of the platoon are obtained by numerically evaluating the eigenvalues of the matrices  $A_{L-F}$  and  $A_L$  (defined in (8.4) and (8.5)). The eigenvalues of the PDE are also computed numerically after using a Galerkin method [170]. The figure shows that the two sets of eigenvalues are in good match. In particular, the least stable eigenvalues are well-captured by the PDE. Additional validation appears in the following sections, where we present and compare results for analysis and control design.

## 8.5 Loss of stability margin with symmetric bidirectional control

In this section, we analyze the stability of a discrete platoon by evaluating the eigenvalues of the PDE

$$\left( \frac{\partial^2}{\partial t^2} + b_0 \frac{\partial}{\partial t} - a_0^2 \frac{\partial^2}{\partial x^2} \right) \tilde{v} = 0, \quad (8.18)$$

where  $x \in [0, 2\pi]$ ,  $\rho_0 = \frac{N}{2\pi}$  is the mean density and

$$a_0^2 := \frac{k_0}{\rho_0^2} \quad (8.19)$$

is the wave speed. The PDE corresponds to the platoon with symmetric and constant control gains:  $k_f(x) = k_b(x) \equiv k_0$  and  $b(x) \equiv b_0$ . On taking the Laplace transform, one obtains the characteristic equation

$$s^2 + b_0 s - a_0^2 \lambda = 0, \quad (8.20)$$

where  $\lambda$  is an eigenvalue of the Laplacian, i.e.,

$$\frac{d^2 \eta}{dx^2} = \lambda \eta(x), \quad (8.21)$$

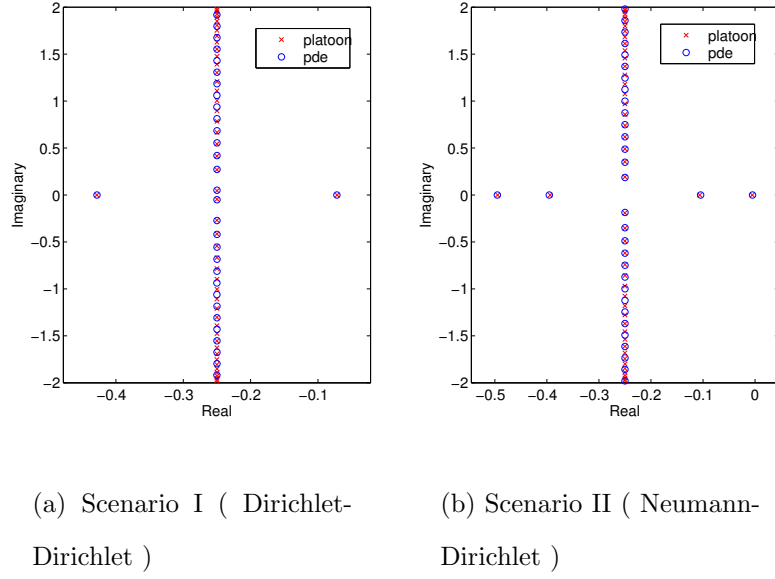


Figure 8.2. Comparison of closed loop eigenvalues of the platoon dynamics and the spectrum of the corresponding PDE (8.18) for the two different scenarios: (a) platoon with fictitious lead and follow vehicles, and correspondingly the PDE (8.18) with Dirichlet boundary conditions, (b) platoon with fictitious lead vehicle, and correspondingly the PDE (8.18) with Neumann-Dirichlet boundary conditions. For ease of comparison, only a few of the eigenvalues are shown. Both plots are for  $N = 25$  vehicles; the controller parameters are  $k_i^{(f)} = k_i^{(b)} = 1$  and  $b_i = 0.5$  for  $i = 1, 2, \dots, N$ , and for the PDE  $k_f(x) \equiv 1$  and  $b(x) \equiv 0.5$ .

<i>boundary condition</i>	<i>eigenvalue <math>\lambda_l</math></i>	<i>eigenfunction <math>\psi_l(x)</math></i>	<i><math>l</math></i>
$\eta(0) = \eta(2\pi) = 0$ ( <i>Dirichlet - Dirichlet</i> )	$-\frac{l^2}{4}$	$\sin(\frac{lx}{2})$	$l = 1, 2, \dots$
$\frac{\partial \eta}{\partial x}(0) = \eta(2\pi) = 0$ ( <i>Neumann - Dirichlet</i> )	$-\frac{(2l-1)^2}{16}$	$\cos(\frac{(2l-1)x}{4})$	$l = 1, 2, \dots$

Table 8.2. The eigen-solutions for the Laplacian with two different boundary conditions.

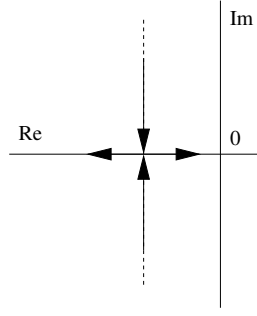
and  $\eta$  is an eigenfunction satisfying appropriate boundary conditions – (8.15) for scenario I and (8.16) for scenario II. The eigen-solutions for the two scenarios are given by the following simple Lemma.

**Lemma 8.5.1.** *Consider the eigenvalue problem (8.21) for the Laplacian with boundary conditions (8.15) and (8.16) corresponding to the scenarios I and II respectively. The eigenvalues and the eigenfunctions for the two scenarios are given in the Table 8.2. The eigenfunctions for either scenario provide a basis of  $L^2([0, 2\pi])$ .*  $\square$

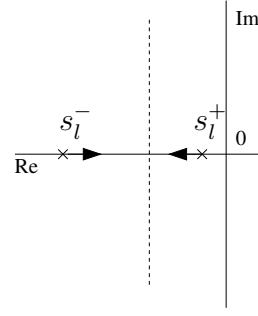
*Proof.* It is a simple calculation to verify that the eigenvalues and eigenfunctions given in the table satisfy the eigenvalue problem. Any eigenfunctions of the Laplacian on  $[0, 2\pi]$  are known to provide a basis for  $L^2([0, 2\pi])$  [171].  $\blacksquare$

To see the effect of  $N$  on stability, we evaluate the eigenvalues (roots of the characteristic equation (8.20)) for the Dirichlet boundary conditions (scenario I). Using Table 8.2, the  $l^{\text{th}}$  eigenvalue is given by

$$s_l^{\pm} = \frac{-b_0 \pm \sqrt{b_0^2 - a_0^2 l^2}}{2}, \quad (8.22)$$



(a) Eigenvalues move toward zero with increasing  $N$ .



(b) Mistuning “exchanges” stability between  $s_l^+$  and  $s_l^-$ .

Figure 8.3. A schematic explaining the loss of stability as  $N$  increases and how mistuning ameliorates this loss.

where  $l = 1, 2, \dots$ . The real part of the eigenvalue depends upon the discriminant  $D(l, N) = (b_0^2 - a_0^2 l^2)$ , where the wave speed  $a_0$  depends both on control gain  $k_0$  and number of vehicles  $N$  (see (8.19)). For a fixed control gain, there are two cases to consider:

1. If  $D(l, N) < 0$ , the roots  $s_l^\pm$  are complex with the real part given by  $-b_0$ ,
2. If  $D(l, N) > 0$ , the roots  $s_l^\pm$  are real with  $s_l^+ + s_l^- = -2b_0$ .

In the former case, the damping is determined by the velocity feedback term  $b_0 \frac{\partial}{\partial t}$ , while in the latter case one eigenvalue ( $s_l^-$ ) gains damping at the expense of the other ( $s_l^+$ ) which loses damping. When  $s_l^\pm$  are real, the eigenvalue  $s_l^+$  is closer to the origin than  $s_l^-$ ; so we call  $s_l^+$  the  $l^{\text{th}}$  *less-stable* eigenvalue. The following lemma gives the dependence of this eigenvalue on the number of vehicles  $N$ .

**Lemma 8.5.2.** *Consider the eigenvalue problem for the linear PDE (8.18) with boundary conditions (8.15) and (8.16), corresponding to scenarios I and II respec-*

boundary condition	$s_l^+$ for $l \ll l_c$	$l_c$
Dirichlet-Dirichlet	$-\frac{\pi^2 k_0}{b_0} \frac{l^2}{N^2} + O(\frac{1}{N^4})$	$\frac{b_0 N}{2\pi\sqrt{k_0}}$
Neumann-Dirichlet	$-\frac{\pi^2 k_0}{4b_0} \frac{l^2}{N^2} + O(\frac{1}{N^4})$	$\frac{b_0 N}{2\pi\sqrt{k_0}}$

Table 8.3. the trend of the less stable eigenvalue  $s_l^+$  for the PDE (8.18)

tively. The  $l^{\text{th}}$  less-stable eigenvalue  $s_l^+$  approaches 0 as  $O(1/N^2)$  in the limit as  $N \rightarrow \infty$ . The asymptotic formulas appear in Table 8.3.  $\square$

*Proof.* We first consider scenario I with Dirichlet boundary conditions (8.15). Using (8.22) and (8.19),

$$2s_l^\pm = -b_0 \pm b_0 \left(1 - \frac{a_0^2 l^2}{b_0^2}\right)^{1/2} = -b_0 \pm b_0 \left(1 - \frac{2\pi^2 k_0}{b_0^2} \frac{l^2}{N^2}\right) + O\left(\frac{1}{N^4}\right)$$

for  $a_0^2 l^2 / b_0^2 \ll 1$ . The asymptotic formula holds for wave numbers

$$l \ll \frac{b_0}{a_0} = \frac{b_0 N}{2\pi\sqrt{k_0}} =: l_c, \quad (8.23)$$

and in particular for each  $l$  as  $N \rightarrow \infty$ . The proof for the scenario II with Neumann-Dirichlet boundary conditions (8.16) follows similarly.  $\blacksquare$

Figure 8.3(a) graphically illustrates the destabilization by depicting the movement of eigenvalues  $s_1^\pm$  as  $N$  increases. For sufficiently small values of  $N$ , the discriminant  $D(1, N)$  is negative and the eigenvalue  $s_1^\pm$  are complex. The real part of the eigenvalue depends only on the value of  $b_0$ . At a critical value of  $N = N_c := \frac{\pi\sqrt{2k_0}}{b_0}$ , the discriminant becomes zero,  $s_1^+ = s_1^-$  and the eigenvalues collide on the real axis. For values of  $N > N_c$  and in particular as  $N \rightarrow \infty$ , the eigenvalue  $s_1^+$  asymptotes to 0 while staying real, and  $s_1^-$  asymptotes to  $-b$ . Their cumulative damping, as reflected in the sum  $s_l^+ + s_l^- = -2b_0$ , is conserved. In

other words,  $s_1^+$  is destabilized at the expense of  $s_1^-$ . The Lemma shows that the *least stable eigenvalue* admits an asymptotic expansion

$$s_1^+ = -\frac{\pi^2 k_0}{b_0} \frac{1}{N^2} + O\left(\frac{1}{N^4}\right) \quad (\text{Dirichlet-Dirichlet}) \quad (8.24)$$

$$= -\frac{\pi^2 k_0}{4b_0} \frac{1}{N^2} + O\left(\frac{1}{N^4}\right) \quad (\text{Neumann-Dirichlet}) \quad (8.25)$$

as  $N \rightarrow \infty$ . Therefore, for large values of  $N$ , the *least-stable* eigenvalue asymptotes to zero as  $O(1/N^2)$ .

Finally, we present numerical computations that corroborates this PDE-based analysis. Figure 8.4 plots as a function of  $N$  the least stable eigenvalue of the PDE and of the discrete platoon as well as the prediction from the asymptotic formula. The eigenvalues for the discrete platoon are obtained by numerically evaluating the eigenvalues of the matrices  $A_{L-F}$  and  $A_L$  (see (8.4) and (8.5)) with constant control gains  $k_i^{(f)} = k_i^{(b)} = k_0 = 1$  and  $b_i = b_0 = 0.5$  for  $i = 1, \dots, N$ . The comparison shows that the PDE analysis closely matches the eigenvalue of the discrete platoon.

*Remark 8.5.1.* : The preceding analysis shows that the loss of stability experienced with a symmetric bidirectional architecture is controller independent. The least stable eigenvalue approaches 0 as  $O(1/N^2)$  irrespective of the values of the gains  $k_0$  and  $b_0$ , as long as they are fixed constants independent of  $N$ . Equations (8.24) and (8.25) also imply that for the least stable eigenvalue to be uniformly bounded away from 0, one has to increase the control gain  $k_0$  as  $N^2$ . This is consistent with the conclusion of Jovanović *et. al.*, who studied the LQR control problem of a platoon on a circle [158].

*Remark 8.5.2.* : The result of Lemma 8.5.2 also indicates that the platoon with both a fictitious leader and a follower (scenario I) has a higher stability margin than platoon with only a fictitious leader (scenario II). We note, however, that in

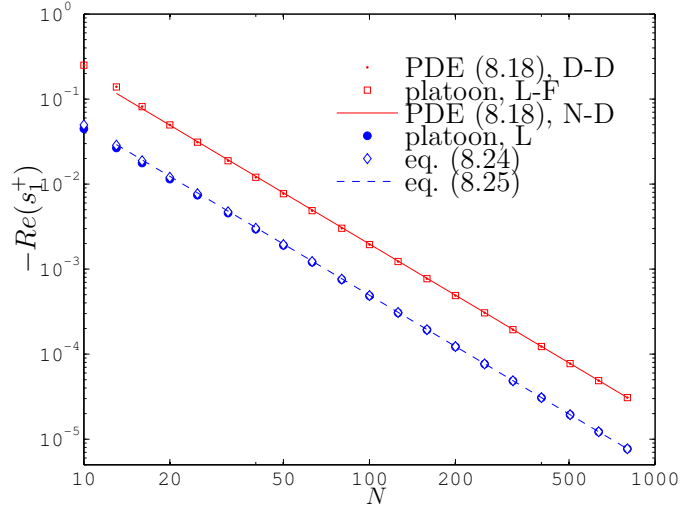


Figure 8.4. Comparison of the least stable eigenvalue of the closed loop platoon dynamics and that predicted by Lemma 8.5.2 with symmetric bidirectional control. In the plot legends, “D-D” stands for “Dirichlet-Dirichlet”, “N-D” for “Neumann-Dirichlet”, “L-F” for fictitious leader-follower, and “L” for fictitious leader. The plot for “PDE (8.18), D-D” should be compared with “platoon, L-F” since they both correspond to scenario I. Similarly, “PDE (8.18), N-D” and “platoon, L-F” correspond to scenario II. Note that the predictions (8.24) and (8.25) are valid for  $1 \ll l_c$  (defined in (8.23)), which in this case means for  $N \gg 12$ .

scenario I, the absolute position of the fictitious follow vehicle must be provided to the last vehicle in the platoon. Therefore scenario I requires one extra piece of global information as compared to the scenario II. The result provides a numerical measure of the benefit of this extra information – a factor of 4 improvement in the closed-loop damping.

## 8.6 Reducing loss of stability by mistuning

With symmetric bidirectional control,  $k_p^{(-)}(x) \equiv 0$ , so the only term left in the right hand side of the governing PDE (8.14) is a  $O(\frac{1}{N^2})$  term. This explains the decay of the least stable eigenvalue as  $1/N^2$ . Any amount of asymmetry between the front and the back gain functions  $k_f(s)$  and  $k_b(x)$  will make  $k_p^{(-)}(x)$  not identically zero, so the right hand side will contain a  $O(\frac{1}{N})$  term. This gives us hope that the least stable eigenvalue might decay less slowly in the presence of such asymmetry, no matter how small. We will now show that this is indeed the case, and determine the gain profiles that achieves this slower rate of decay.

We consider the eigenvalue problem for the PDE (8.14) where the control gains are designed (mistuned) with the objective of minimizing the least-stable eigenvalue  $s_1^+$ . In particular, we consider forward and backward position feedback gain profiles:

$$\begin{aligned} k_f(x) &= k_0 + \epsilon k_f^a(x), \\ k_b(x) &= k_0 + \epsilon k_b^a(x), \end{aligned}$$

where  $\epsilon > 0$  is a small parameter signifying the amount of mistuning and  $k_f^a(x)$ ,

$k_b^a(x)$  denote the perturbation profiles. Define

$$\begin{aligned} k_s(x) &:= k_f^a(x) + k_b^a(x), \\ k_m(x) &:= k_f^a(x) - k_b^a(x), \end{aligned}$$

so that from (8.12),

$$\begin{aligned} k_p^{(+)}(x) &= 2k_0 + \epsilon k_s(x), \\ k_p^{(-)}(x) &= \epsilon k_m(x). \end{aligned}$$

The mistuned version of the PDE (8.14) is then given by

$$\frac{\partial^2 \tilde{v}}{\partial t^2} + b_0 \frac{\partial \tilde{v}}{\partial t} = L(\epsilon) \tilde{v}, \quad (8.26)$$

where

$$L(\epsilon) \tilde{v} := a_0^2 \frac{\partial^2 \tilde{v}}{\partial x^2} + \epsilon \left[ \frac{k_m}{\rho_0} \frac{\partial \tilde{v}}{\partial x} + \frac{1}{2\rho_0^2} \frac{\partial}{\partial x} \left( k_s \frac{\partial \tilde{v}}{\partial x} \right) \right] \quad (8.27)$$

In the remainder of this chapter, we study the problem of optimizing the stability margins by judicious choice of  $k_m(x)$  and  $k_s(x)$ . In effect, the results of our investigation, carried out in the following two sections using perturbation and optimization methods, provide a systematic framework for designing control gains in the discrete platoon.

### 8.6.1 Perturbation analysis

The control objective is to design mistuning profiles  $k_m(x)$  and  $k_s(x)$  to *minimize* the least stable eigenvalue  $s_1^+$ . To achieve this, we first use a perturbation method, borrowed from [168], to obtain an explicit asymptotic formula for the eigenvalues.

**Theorem 8.6.1.** *Consider the eigenvalue problem for the mistuned PDE (8.26) with Dirichlet boundary condition (8.15) corresponding to scenario I. The  $l^{\text{th}}$  eigenvalue pair is given by the asymptotic formula*

$$\begin{aligned} s_l^+(\epsilon) &= \epsilon \frac{l}{2b_0 N} \int_0^{2\pi} k_m(x) \sin(lx) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right), \\ s_l^-(\epsilon) &= -2b_0 - \epsilon \frac{l}{2b_0 N} \int_0^{2\pi} k_m(x) \sin(lx) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right), \end{aligned}$$

that is valid for each  $l$  in the limit as  $\epsilon \rightarrow 0$  and  $N \rightarrow \infty$ .  $\square$

The perturbation formula is useful because it suggests the most beneficial mistuning profile  $k_m(x)$  in the limit  $\epsilon \rightarrow 0$ , which is summarized in the next corollary.

**Corollary 8.6.1.** *Consider the problem of minimizing the least-stable eigenvalue of the PDE (8.26) with Dirichlet boundary condition (8.15) by choosing a function  $k_m(x) \in L^2([0, 2\pi])$  with norm-constraint  $\int_0^{2\pi} |k_m(x)|^2 dx = 1$ . In the limit as  $\epsilon \rightarrow 0$ , the optimal mistuning profile is given by  $k_m(x) = -\frac{\sin(x)}{\sqrt{\pi}}$ . With this profile, the least stable eigenvalue is given by the asymptotic formula*

$$s_1^+(\epsilon) = -\frac{\epsilon\sqrt{\pi}}{2b_0} \frac{1}{N} + O(\epsilon^2) + O\left(\frac{1}{N^2}\right)$$

in the limit as  $\epsilon \rightarrow 0$  and  $N \rightarrow \infty$ .  $\square$

This result shows that even with an *arbitrarily small amount* of mistuning  $\epsilon$ , one can improve the closed-loop platoon damping by a large amount, especially for large values of  $N$ . The least-stable eigenvalue  $s_1^+$  asymptotes to 0 as  $O(\frac{1}{N})$  in the mistuned case as opposed to  $O(\frac{1}{N^2})$  in the nominal case.

Figure 8.3(b) graphically illustrates the mechanism by which mistuning affects the movement of eigenvalues  $s_1^\pm$  as  $N$  increases. By properly choosing the

mistuning patterns  $k_m(x)$  and  $k_s(x)$ , the damping can be exchanged between the eigenvalues  $s_1^+$  and  $s_1^-$  so that the less stable eigenvalue  $s_1^+$  “gains” stability at the expense of the more stable eigenvalue  $s_1^-$ . The net amount of damping is preserved, since  $s_1^+ + s_1^- = 0$  (as seen from Theorem 8.6.1).

Figure 8.5 presents a validation of these results by comparing the numerically obtained mistuned and nominal eigenvalues for both the discrete platoon and the PDE; the parameter values are indicated in the figure caption. The figure shows that

1. the platoon eigenvalues match the PDE eigenvalues accurately over a range of  $N$ , and
2. the mistuned eigenvalues show an order of magnitude improvement over the nominal case even though the controller gains differ from their nominal values only by  $\pm 10\%$ .

For comparison, the figure also depicts the asymptotic eigenvalue formula given in Corollary 8.6.1. Similar results are also obtained for scenario II, which are summarized by the next theorem.

**Theorem 8.6.2.** *Consider the eigenvalue problem for the mistuned PDE (8.26) with Neumann-Dirichlet boundary condition (8.16) corresponding to scenario II. The  $l^{\text{th}}$  eigenvalue pair is given by the asymptotic formula*

$$\begin{aligned} s_l^+(\epsilon) &= -\epsilon \frac{l}{4b_0N} \int_0^{2\pi} k_m(x) \sin\left(\frac{lx}{2}\right) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right), \\ s_l^-(\epsilon) &= -2b_0 + \epsilon \frac{l}{4b_0N} \int_0^{2\pi} k_m(x) \sin\left(\frac{lx}{2}\right) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right), \end{aligned}$$

that is valid for each  $l$  in the limit as  $\epsilon \rightarrow 0$  and  $N \rightarrow \infty$ . □

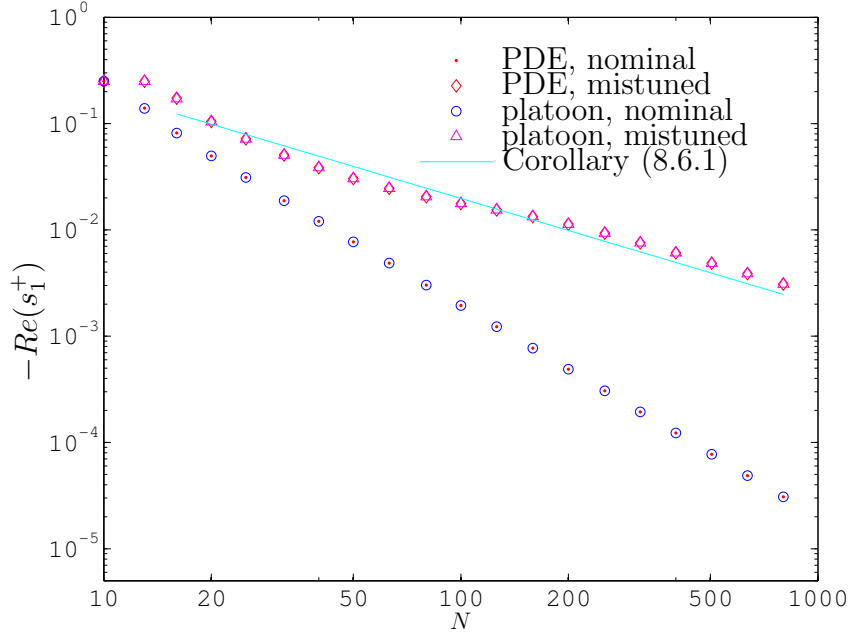


Figure 8.5. The least stable eigenvalue of the closed loop platoon (i.e., of  $A_{L-F}$  in (8.4)) and of the PDE (8.26) with Dirichlet boundary conditions, with and without mistuning, for a range of values of  $N$ . Parameters for the nominal case are  $k_0 = 1$  and  $b_0 = 0.5$ . In the mistuned case, forward and backward controller gains are chosen as  $k_f = k_0 - 0.1 \sin(x)$ ,  $k_b = k_0 + 0.1 \sin(x)$  (i.e.,  $k_m(x) = -\sin(x)/\sqrt{\pi}$ ,  $k_s(x) \equiv 0$  and  $\epsilon = 0.2\sqrt{\pi}$ ). For the platoon, controller gains are chosen by sampling the gains for the PDE as  $k_i^{(f)} = 1 - 0.1 \sin(y_i^d)$ ,  $k_i^{(b)} = 1 + 0.1 \sin(y_i^d)$ , where  $y_i^d$  defined in (8.2) is the desired position of the  $i^{\text{th}}$  vehicle. The legend “Corollary 8.6.1” refers to the prediction by Corollary 8.6.1. Note that the prediction of Corollary 8.6.1 is plotted only for  $N > 16$  to ensure that  $1 \ll l_c$  (see (8.23)).

As with scenario I, here again we use the above result to determine the most beneficial profile  $k_m(x)$  for small  $\epsilon$ :

**Corollary 8.6.2.** *Consider the problem of minimizing the least-stable eigenvalue of the PDE (8.26) with Neumann-Dirichlet boundary conditions (8.16) by choosing a function  $k_m(x) \in L^2([0, 2\pi])$  with norm-constraint  $\int_0^{2\pi} |k_m(x)|^2 dx = 1$ . In the limit as  $\epsilon \rightarrow 0$ , the optimal mistuning profile is given by  $k_m(x) = \frac{1}{\sqrt{\pi}} \sin(\frac{x}{2})$ . With this profile, the least-stable eigenvalue is given by the asymptotic formula*

$$s_1^+(\epsilon) = -\frac{\epsilon\sqrt{\pi}}{4b_0} \frac{1}{N} + O(\epsilon^2) + O(\frac{1}{N^2})$$

in the limit as  $\epsilon \rightarrow 0$  and  $N \rightarrow \infty$ . □

Eigenvalue trends together with their validation for scenario II appear in Fig. 8.6.

The proof of Theorem 8.6.1 is presented below. The proof of Theorem 8.6.2 is analogous and is therefore omitted.

*Proof of Theorem 8.6.1.* The spatial inhomogeneity introduced by the  $x$ -dependent coefficients  $k_m(x)$  and  $k_s(x)$  destroy the spatial invariance of the nominal PDE (8.18). Hence, the Fourier basis – eigenfunctions of the Laplacian – no longer lead to a diagonalization of the mistuned PDE. The methods of section 8.5 thus need to be suitably modified. In order to compute the eigenvalues for the mistuned PDE (8.26), we take a Laplace transform of (8.26) and get

$$-a_0^2 \frac{\partial^2 \eta}{\partial x^2} + s^2 \eta + b_0 s \eta = \epsilon \left[ \frac{k_m}{\rho_0} \frac{\partial \eta}{\partial x} + \frac{1}{2\rho_0^2} \frac{\partial}{\partial x} (k_s \frac{\partial \eta}{\partial x}) \right]. \quad (8.28)$$

We are interested in eigenvalues of (8.28) with Dirichlet boundary conditions, i.e., the values of  $s$  for which a solution to the homogeneous PDE (8.28) exists with

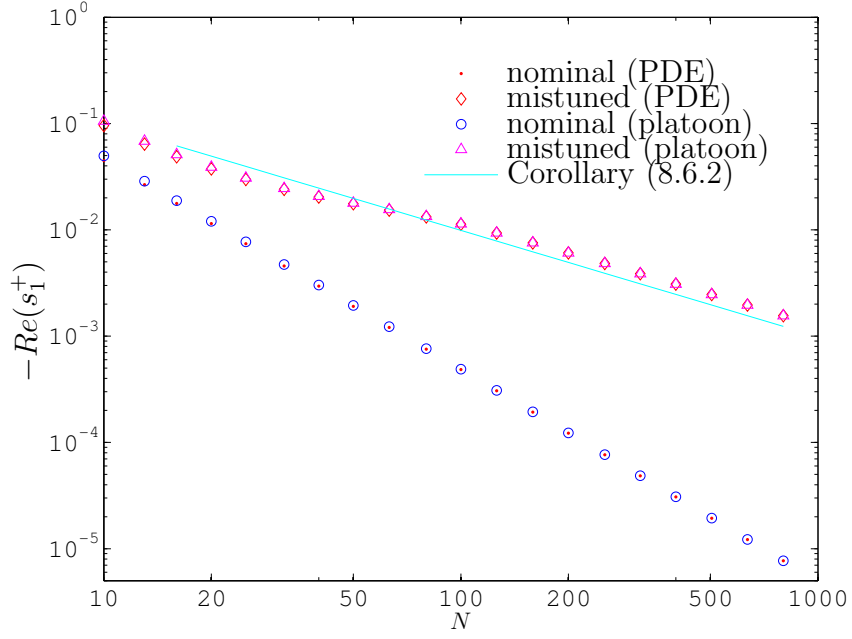


Figure 8.6. The least stable eigenvalue of the closed loop platoon in scenario II (i.e., of  $A_L$  in (8.4)) and of the PDE (8.26) with Neumann-Dirichlet b.c., with and without mistuning, for a range of values of  $N$ . Parameters for the nominal case are  $k_0 = 1$ ,  $b_0 = 0.5$ . In the mistuned case, forward and backward controller gains are chosen as  $k_f = k_0 + 0.1 \sin(\frac{x}{2})$ ,  $k_b = k_0 - 0.1 \sin(\frac{x}{2})$  (i.e.,  $k_m = \frac{1}{\sqrt{\pi}} \sin(\frac{x}{2})$ ,  $k_s(x) \equiv 0$  and  $\epsilon = 0.2\sqrt{\pi}$ . For the platoon, the gains are chosen as  $k_i^{(f)} = k_0 + 0.1 \sin(y_i^d/2)$  and  $k_i^{(b)} = k_0 - 0.1 \sin(y_i^d/2)$ , where  $y_i^d$  defined in (8.2) is the desired position of the  $i^{\text{th}}$  vehicle. The legend “Corollary 8.6.2” refers to the prediction by Corollary 8.6.2 of mistuned PDE eigenvalues.

boundary conditions  $\eta(0) = \eta(2\pi) = 0$ . To obtain these eigenvalues, we use a perturbation method expressing the eigenfunction and eigenvalue in a series form:

$$\eta(x) = \eta_0(x) + \epsilon\eta_1(x) + O(\epsilon^2), \quad (8.29)$$

$$s = r_0 + \epsilon r_1 + O(\epsilon^2). \quad (8.30)$$

We note that  $\epsilon r_1$  denotes the perturbation to the nominal eigenvalue  $r_0$  as a result of the mistuning. Substituting (8.30) in (8.28) and doing an  $O(1)$  balance, we get

$$O(1) : -a_0^2(\eta_0)_{xx} + r_0^2\eta_0 + br_0\eta_0 = 0, \quad (8.31)$$

whose eigen-solution is given by

$$\eta_0 = d_l \sin\left(\frac{lx}{2}\right), \quad (8.32)$$

$$r_0 = s_l^\pm(0), \quad (8.33)$$

where  $l = 1, 2, \dots$ ,  $d_l$  is an arbitrary real constant, and  $s_l^\pm(0)$  is given by (8.22).

Next,

$$\begin{aligned} O(\epsilon) : -a_0^2(\eta_1)_{xx} + (r_0^2 + br_0)\eta_1 &= +\frac{k_m}{\rho_0}\frac{\partial\eta_0}{\partial x} + \frac{1}{2\rho_0^2}\frac{\partial}{\partial x}\left(k_s\frac{\partial\eta_0}{\partial x}\right) \\ &\quad - (2r_0r_1 + b_0r_1)\eta_0 \\ &:= R \end{aligned} \quad (8.34)$$

Substituting  $r_0 = s_l^\pm(0)$  on the left hand side leads to a resonance condition for the right hand side term, denoted by  $R$ . In particular for a solution  $\eta_1$  to exist,  $R$  must lie in the range space of the linear operator

$$\left(-a_0^2\frac{\partial^2}{\partial x^2} + (r_0^2 + br_0)\right). \quad (8.35)$$

For this self-adjoint operator, the range space is the complement of its null space  $\{\sin(\frac{lx}{2})\}$ . This gives the resonance condition as

$$\frac{1}{\pi} < R, \sin\left(\frac{lx}{2}\right) > = 0,$$

where  $\langle \cdot, \cdot \rangle$  denotes the standard inner product in  $L^2(0, 2\pi)$ . Explicitly, this leads to an equation

$$(2r_0 + b_0)r_1 = \frac{l}{4\pi\rho_0} \int_0^{2\pi} k_m(x) \sin(lx) dx + \frac{l^2}{8\pi\rho_0^2} \int_0^{2\pi} k_s(x) \cos^2\left(\frac{lx}{2}\right) dx \quad (8.36)$$

For values of  $r_0 = s_l^\pm(0)$ , where  $s_l^\pm(0)$  is given by (8.22), the equation above leads to an expression for perturbation in the two eigenvalues. We denote these perturbations as  $r_1^\pm$ . For  $r_0 = s_k^+(0)$ , we have from Lemma (8.5.2) that  $b_0 \gg |2r_0|$  when  $l \ll l_c$ , so that

$$r_1^+ \approx \frac{l}{4\pi\rho_0 b_0} \int_0^{2\pi} k_m(x) \sin(lx) dx + O\left(\frac{1}{N^2}\right). \quad (8.37)$$

Note that we have dropped the second integral on the right hand side of (8.36) because  $\frac{1}{\rho_0^2} = O(1/N^2)$  for large  $N$ . For  $r_0 = s_k^-(0)$ ,  $2r_0 \approx -2b_0$  for  $l \ll l_c$  and

$$r_1^- \approx -\frac{l}{4\pi\rho_0 b_0} \int_0^{2\pi} k_m(x) \sin(lx) dx + O\left(\frac{1}{N^2}\right). \quad (8.38)$$

Note that

$$r_1^+ + r_1^- = 0.$$

Putting the formulas for the perturbation to the eigenvalues (8.37) and (8.38) in (8.30), we get

$$\begin{aligned} s_l^+(\epsilon) &\approx s_l^+(0) - \epsilon \frac{l}{4\pi b_0 \rho_0} \int_0^{2\pi} k_m(x) \sin(lx) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right), \\ s_l^-(\epsilon) &\approx -2b_0 + \epsilon \frac{l}{4\pi b_0 \rho_0} \int_0^{2\pi} k_m(x) \sin(lx) dx + O(\epsilon^2) + O\left(\frac{1}{N^2}\right). \end{aligned}$$

Since  $s_l^\pm(0) = O(\frac{1}{N^2})$  for  $l < l_c$  (Lemma 8.5.2) and  $\rho_0 = \frac{N}{2\pi}$ , the result follows.

*Remark 8.6.1.* The asymptotic formulae for  $s_l^\pm$  in Corollary 8.6.1 and Corollary 8.6.2 are valid only in the limit  $\epsilon \rightarrow 0$ . However, one would like to be able to use them with somewhat larger values of  $\epsilon$  to realize the benefit of mistuning.

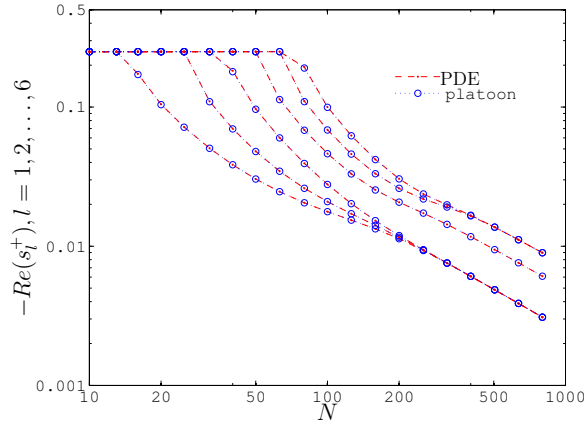


Figure 8.7. The real parts of six eigenvalues (closest to 0) of the closed loop platoon dynamics for Scenario I, and their comparison with the PDE eigenvalues with Dirichlet-Dirichlet boundary conditions, for mistuned gains  $k_f(x) = 1 - 0.1 \sin(x)$ ,  $k_b(x) = 1 + 0.1 \sin(x)$  and  $b(x) \equiv 0.5$ . As predicted by the S-L theory, the least stable eigenvalue stays the least stable, although eigenvalues that were more stable merge with it as  $N$  increases.

To do so, one has to preclude the possibility of “eigenvalue cross-over,” i.e., of the second ( $s_2^+$ ) or some other marginally stable eigenvalue from becoming the least stable eigenvalue in the presence of mistuning. It turns out that such a cross-over is ruled out as a consequence of the Strum-Liouville (S-L) theory for the elliptic boundary value problems. The standard argument relies on the positivity of the eigenfunction corresponding to  $s_1^+$ ; the reader is referred to [171] for the details. Figure 8.7 verifies this numerically by depicting the six eigenvalues closest to 0 (for both the PDE and the discrete platoon) as a function of  $N$  when mistuning is applied.

## 8.6.2 Eigenvalue optimization

For relatively large values of  $\epsilon$ , the mistuning profiles obtained in the previous section may not be optimal. One therefore needs to find the optimal mistuning gain profiles that minimize the least stable eigenvalue of the PDE (8.26). It can be shown in a straightforward manner that the least-stable eigenvalue of the PDE (8.26) has the following property:

$$s_1^+ \rightarrow \frac{\lambda_1}{b_0} \text{ as } N \rightarrow \infty, \quad (8.39)$$

where  $\lambda_1$  is the principal (with largest real part) eigenvalue of  $L(\epsilon)$  (defined in (8.27)). Thus, in the limit of large number of vehicles, the problem of minimizing the least stable eigenvalue of the PDE (8.26) is equivalent to minimizing  $\lambda_1$ , the principal (with largest real part) eigenvalue of  $L(\epsilon)$  by choosing the functions  $k_m(s), k_s(x) \in L^2$ . By a standard argument in S-L theory,  $\lambda_1$  is real with a positive eigenfunction [171].

For problem of minimizing  $\lambda_1$  to be well-posed, an additional constraint on  $k_m(x)$  and  $k_s(x)$  is needed. In the following, we assume  $k_s(x) \equiv 0$  and impose a constraint

$$\int_0^{2\pi} |k_m(x)|^2 dx = 1. \quad (8.40)$$

$k_s(x) = 0$  is assumed for the sake of simplicity of the presentation and because it appears as a coefficient  $\frac{k_s(x)}{N^2}$ . Any improvement due to  $k_s(x)$  alone is  $O(\frac{1}{N^2})$  while  $k_m(x)$  can potentially deliver an  $O(\frac{1}{N})$  shift in eigenvalue location. This is also reflected in estimates obtained using the perturbation methods (see (8.36)). Thus, the problem of minimizing the least stable eigenvalue of the PDE (8.26) is converted to the following optimization problem:

$$\min_{\{k_s(x) \equiv 0, \int_0^{2\pi} |k_m(x)|^2 dx = 1\}} \lambda_1. \quad (8.41)$$

Even with  $k_m(x)$  alone, the optimization of a non-symmetric eigenvalue problem (as in our case) is challenging with limited theory for guidance; see section 16 of the review paper [172] on eigenvalue optimization. As a first step, we relax the optimization problem by replacing the operator  $L(\epsilon)$  by its symmetric component:

$$L^s(\epsilon)\eta = \left(\frac{L + L^*}{2}\right)\eta = a_0^2 \frac{d^2\eta}{dx^2} - \frac{\epsilon}{2\rho_0} k'_m(x)\eta, \quad (8.42)$$

where  $L^*$  is the adjoint of  $L$  and  $k'_m(x) := \frac{dk_m}{dx}(x)$ . Let  $\lambda_1^s$  denote the principal eigenvalue of  $L^*$ . The following lemma gives the relationship between  $\lambda_1$  and  $\lambda_1^s$ .

**Lemma 8.6.1.** *Let  $\lambda_1$  denote the principal eigenvalue of the operator  $L$  in (8.27) and  $\lambda_1^s$  denote the principal eigenvalue of the symmetric operator  $L^s$  in (8.42). Then*

$$\lambda_1 \leq \lambda_1^s. \quad \square$$

*Proof.* Let  $\lambda_1$  be the principal eigenvalue and  $\phi(x)$  be the corresponding positive eigenfunction of the non-symmetric problem:

$$a_0^2 \frac{d^2\phi}{dx^2} + \epsilon \frac{k_m(x)}{\rho_0} \frac{\partial\phi}{\partial x} = \lambda_1 \phi.$$

Multiplying by  $\phi$  and integrating by parts, we obtain

$$-a_0^2 \int_0^{2\pi} \left(\frac{d\phi}{dx}\right)^2 dx - \frac{\epsilon}{2\rho_0} \int_0^{2\pi} k'_m(x) \phi^2 dx = \lambda_1 \int_0^{2\pi} \phi^2 dx.$$

We have

$$\lambda_1 \leq \max_{\phi > 0} \frac{\left[-a_0^2 \int \left(\frac{d\phi}{dx}\right)^2 dx - \frac{\epsilon}{2\rho_0} \int k'_m(x) \phi^2 dx\right]}{\int \phi^2 dx} = \lambda_1^s,$$

where the last equality follows from the variational characterization of the principal eigenvalue for a symmetric elliptic problem. ■

Instead of the original eigenvalue optimization problem (8.41), we pose and solve the following simpler optimization problem:

$$\min_{\{k_m(x): \int |k_m|^2 dx = 1\}} \lambda^s \quad (8.43)$$

where  $\lambda^s$  is the principal (largest) eigenvalue of  $L^s$ :

$$a_0^2 \frac{d^2 \phi}{dx^2} - \frac{\epsilon}{2\rho_0} k'_m(x) \phi = \lambda^s \phi \quad (8.44)$$

with Dirichlet boundary conditions (8.16). Because of Lemma 8.6.1, the solution to this problem provides an upper bound on the solution to (8.41). Among the two roots of the characteristic equation  $s^2 + b_0 s - \lambda^s = 0$ , the one closer to 0 – which we denote by  $(s_1^+)^s$  – is the least stable eigenvalue of the mistuned symmetric PDE

$$\frac{\partial^2 \tilde{v}}{\partial t^2} + b_0 \frac{\partial \tilde{v}}{\partial t} = L^*(\epsilon) \tilde{v}. \quad (8.45)$$

In the symmetric and non-symmetric cases respectively,

$$(s_1^+)^s \rightarrow \frac{\lambda^s}{b_0} \text{ as } N \rightarrow \infty, \quad \text{and} \quad s_1^+ \rightarrow \frac{\lambda}{b_0} \text{ as } N \rightarrow \infty.$$

From Lemma 8.6.1, we get  $(s_1^+)^s \leq s_1^+$  in the limit of large  $N$ .

The calculations leading to formulas for the principal eigenvalue, eigenfunction, and the optimal mistuning gain profile for the symmetric PDE are presented in Section 8.9. Figure 8.8 presents the optimum mistuning profiles for the symmetric PDE for three different values of  $\epsilon$ . Before presenting numerical validation of eigenvalues, we summarize the main conclusions of the optimization calculations in Section 8.9:

1. The optimization calculations provide a rigorous  $O(\frac{1}{N})$  bound on the least stable eigenvalue of the symmetric problem for non-vanishing values of mistuning amplitude  $\epsilon$ . Using Lemma 8.6.1, this leads to an  $O(\frac{1}{N})$  bound for the least stable eigenvalue of the non-symmetric pde and hence the discrete platoon.

2. The results of the optimization calculation are also shown to be consistent with the results of perturbation analysis. In particular, the optimal mistuning profile tends to the beneficial sinusoidal profile (see Corollary 8.6.1) in the limit  $\epsilon \rightarrow 0$ .
3. Using a symmetry technique, the optimal mistuning profile for scenario II is shown to be a stretched (by factor of 2) version of the optimal mistuning profile for the scenario I case. A formula linking the two is given which helps generalize the results of perturbation analysis to non-vanishing values of  $\epsilon$ .

Figure 8.9 shows the trend of the least stable eigenvalue of the symmetric PDE with the optimum mistuning profile  $k_m^*(x)$ , and compares it with the least stable eigenvalues of the non-symmetric PDE (8.26) under both sinusoidal and  $k_m^*(x)$  mistuning. It is seen from the figure that with the mistuning profile  $k_m^*(x)$ , the least stable eigenvalue for the non-symmetric PDE is smaller (more to the left) than its symmetric counterpart. This is consistent with the conclusion of Lemma 8.6.1. However, numerically the optimal mistuning profile obtained for symmetric PDE was found to be sub-optimal for the non-symmetric PDE corresponding to the discrete platoon. In particular, for the values of  $\epsilon$  tested and shown in the figures, the sinusoidal mistuning profile was seen to provide greater damping for the discrete platoon. The numerically computed least stable eigenvalue of the symmetric PDE with the optimal mistuning profile matches the formula for the same derived in the appendix and it approaches 0 as  $O(\frac{1}{N})$ ; see Figure 8.9. In all cases, eigenvalues of the PDE (8.26) closely matched the discrete platoon eigenvalues.

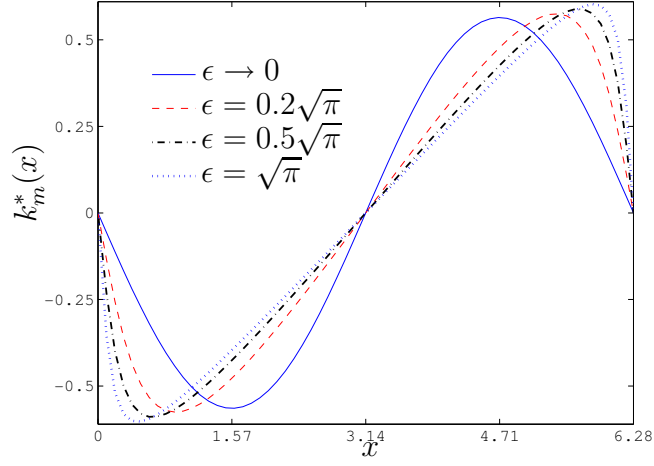


Figure 8.8. The optimal mistuning pattern  $k_m^*(x)$  for the symmetric PDE computed according to the procedure laid out in section 8.6.2 for three different values of  $\epsilon$ . The parameters are  $N = 50$  and  $k_0 = 1$ .

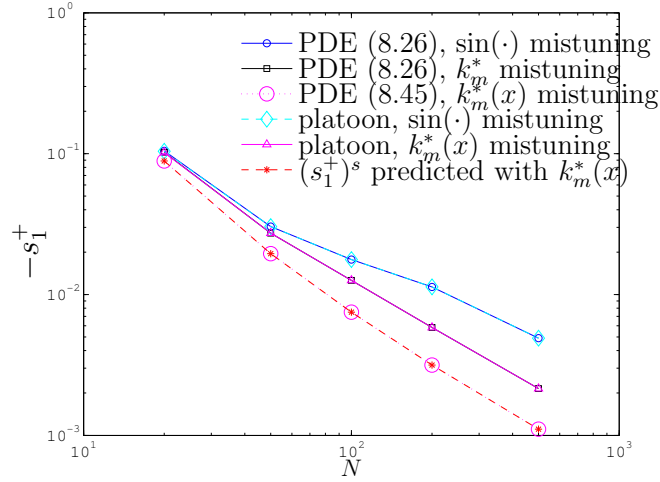


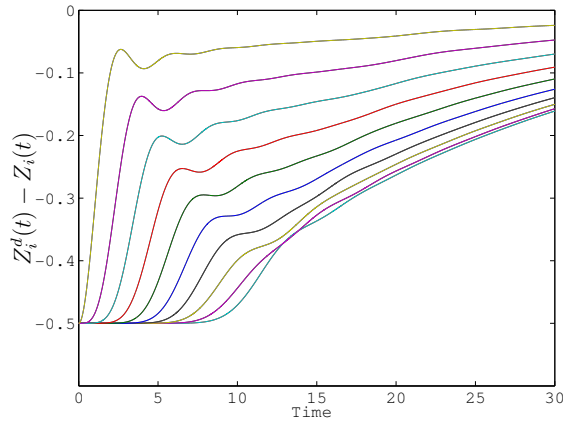
Figure 8.9. The least stable eigenvalue when the optimum mistuning profile computed from the eigenvalue optimization is implemented with  $\epsilon = 0.2\sqrt{\pi}$ . For comparison, several other eigenvalue calculations are shown. PDE (8.45) is the symmetric PDE.

## 8.7 Simulations

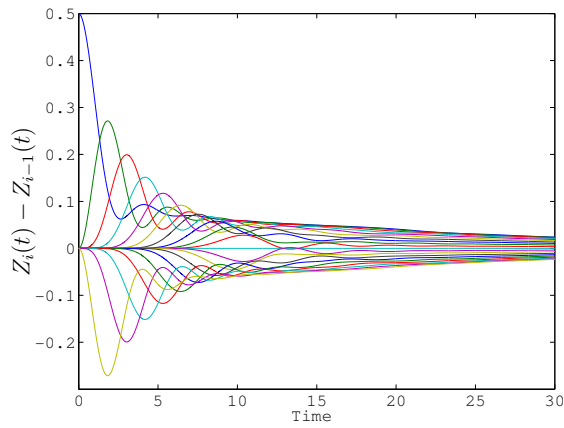
We now present results of a few simulations that show the time-domain improvements – manifested in faster decay of initial errors – with the mistuning-based design of control gains. Simulations were carried out for a platoon of  $N = 20$  vehicles with scenario I, i.e., with fictitious lead and follow vehicles. The desired gap was  $\Delta = 1$  and desired velocity was  $V_d = 5$ . The initial velocity of every vehicle was chosen as the desired velocity and the initial position of the  $i^{\text{th}}$  vehicle was chosen as  $Z_i(0) = i\Delta - 0.5$  for  $i = \{1, \dots, N\}$ .

Figure 8.10 depicts the time-histories of the absolute and relative position errors of the individual vehicles with a symmetric bidirectional control, where the control gains were chosen as  $k_i^{(f)} = k_i^{(b)} = 1$  and  $b_i = 0.5$  for  $i = \{1, \dots, N\}$ . The position errors shown are un-normalized, i.e., absolute position error of the  $i^{\text{th}}$  vehicle is  $Z_i - Z_i^d$  and the relative position error is  $Z_{i-1} - Z_i - \Delta$ .

Figure 8.11 depicts the time-histories of the absolute and relative position errors for the platoon with mistuned controller gains. The mistuning profile was chosen according to Corollary 8.6.1 so that maximum and minimum gains over all vehicles is within  $\pm 10\%$  of the nominal value. On comparing Figures 8.10 and 8.11, we see that the errors in the initial conditions are reduced faster in the mistuned case compared to the nominal case. These observations are consistent with the improvement in the closed-loop stability margin with the mistuned design.

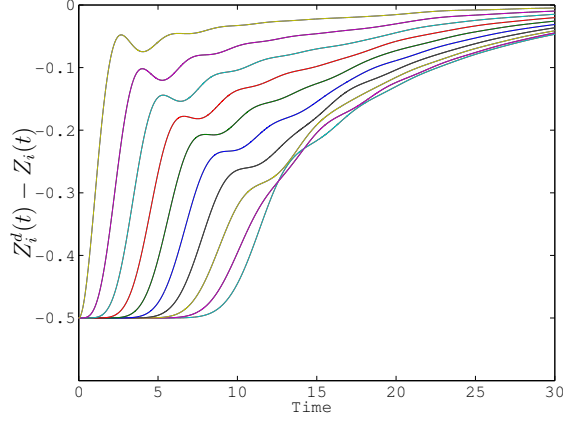


(a) Absolute position errors

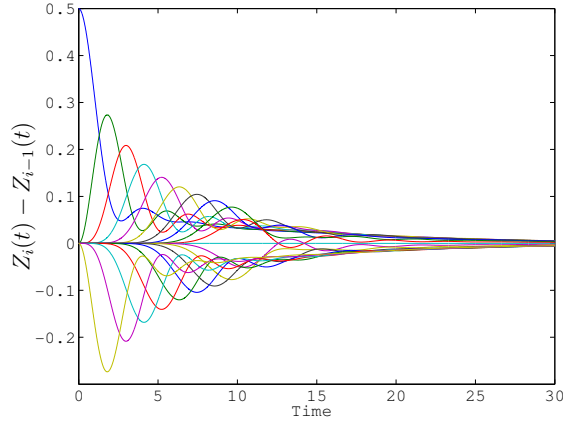


(b) Relative position errors

Figure 8.10. Simulations with symmetric bidirectional control. Time histories of the absolute and relative position errors of the vehicles in a platoon with symmetric bidirectional control (scenario I) are shown.



(a) Absolute position errors.



(b) Relative position errors.

Figure 8.11. Simulations with mistuned bidirectional control. Time histories of the absolute and relative position errors of the vehicles in a platoon (scenario I) with mistuned bidirectional control are shown. Controller gains were chosen as  $k_i^{(f)} = 1 - 0.1 \sin(y_i^d)$ ,  $k_i^{(b)} = 1 + 0.1 \sin(y_i^d)$ , where  $y_i^d$  defined in (8.2).

## 8.8 Comments and open problems

Although the model is derived under the assumption of large number of vehicles  $N$ , in practice it provides quantitatively correct predictions for the discrete platoon dynamics even for relatively small values of  $N$ . The advantage of the PDE formulation is reflected in the ease with which the spectrum can be obtained even with non-symmetric boundary conditions. Finally, certain important aspects such as the beneficial nature of forward-backward asymmetry in control gains as revealed by the PDE is difficult to see with the discrete platoon model.

A promising research direction is exploring the use of PDE-based models for design and analysis of decentralized controllers for a fleet of vehicles in 2 or 3 spatial dimensions. Another promising direction is the study of formations with time-varying topology through a continuum model. Multi-agent coordination problems in which the interconnections between agents may change with time are difficult to analyze due to their time-varying nature. Perhaps an “aggregate” view afforded by a continuum approximation – in the form of an appropriate PDE – can be useful in analysis. However, how to derive the governing PDE for such a situation and how to validate it is not clear.

We did not investigate in this chapter if the mistuning based design reduces the amplification of disturbance that is typically seen in automated platoons. In light of the results in Chapter 7 this is an important problem and needs to be studied in the future.

## 8.9 Proofs

In this section, we present the solution of the eigenvalue optimization problem (8.43). The optimization is based upon a method originally due to Keller [173]. Assume  $k_m^*(x)$  minimizes the largest eigenvalue and is thus the solution. After Keller, we introduce a family of functions  $k_m(x, \delta)$  with  $k_m^*(x) = k_m(x, 0)$  to construct a differential characterization of this optimal. For each  $\delta$ , the principal eigenvalue and the eigenfunction are given by  $\lambda^s(\delta)$  and  $\phi(x, \delta)$  respectively. Differentiating (8.44) with respect to  $\delta$  and evaluating at  $\delta = 0$  gives

$$a_0^2 \frac{d^2 \phi_\delta}{dx^2} - \frac{\epsilon}{2\rho_0} k'_m(x) \phi_\delta - \frac{\epsilon}{2\rho_0} (k'_m)_\delta(x) \phi = \lambda^s \phi_\delta,$$

where  $(k'_m)_\delta(x) = \frac{\partial(k'_m)(x)}{\partial\delta}|_{\delta=0}$  and  $\phi_\delta = \frac{\partial\phi}{\partial\delta}|_{\delta=0}$ . Multiplying by  $\phi$ , integrating, and using (8.44) gives

$$\int_0^{2\pi} (k'_m)_\delta(x) \phi^2 dx = 0 \quad (8.46)$$

On differentiating the constraint (8.40), we obtain

$$\int_0^{2\pi} k_m(x) (k_m)_\delta(x) dx = 0. \quad (8.47)$$

Since  $(k_m)_\delta(x)$  represents an arbitrary perturbation about the optimal, the two equations (8.46)-(8.47) imply that the optimal mistuning pattern is given by

$$k_m^*(x) = -C \phi \frac{d\phi}{dx}, \quad (8.48)$$

where  $C$  is some constant. It follows that

$$k'_m(x) = -\frac{C}{2} \frac{d^2(\phi)^2}{dx^2}$$

and substituting this in (8.44), one obtains a nonlinear BVP

$$a_0^2 \phi'' + \frac{\epsilon C}{2\rho_0} [\phi \phi'' + (\phi')^2] \phi = \lambda^s \phi. \quad (8.49)$$

with Dirichlet boundary conditions (8.16). The principal eigenfunction of this problem then defines the optimal mistuning profile  $k_m^*(x)$  by (8.48).

Before presenting the details of the calculations, we make two observations. One, the nonlinear BVP admits a symmetry whereby if  $\phi(x)$  is an eigenfunction then so is  $\phi(2\pi - x)$ . Now, the principal eigenfunction of any elliptic eigenvalue problem is known to be unique (see Ch. 15 of [174]). This implies that  $\phi(x) = \phi(2\pi - x)$ , and at  $x = \pi$

$$\frac{d\phi}{dx}(\pi) = 0. \quad (8.50)$$

Thus, the solution  $\phi(x)$  of (8.49) also provides for the principal eigenfunction with the Neumann-Dirichlet boundary conditions (8.16). It is given by  $\phi(\frac{x}{2} + \pi)$  and the optimal profile is obtained as before by using (8.48).

The second observation pertains to a comparison with the results obtained using perturbation methods. To do so, we consider the  $\epsilon \rightarrow 0$  limit first. In this limit, the principal eigenfunction (of (8.49)) is given by  $\phi = \sin(\frac{x}{2})$ . Using (8.48), one obtains the optimal mistuning pattern for the limiting case

$$k_m^*(x) = -C \sin(x),$$

where  $C = \frac{1}{\pi}$  satisfies the norm constraint. This is consistent with the optimal mistuning profile obtained using perturbation methods. For small  $\epsilon$ , this also provides an estimate of the eigenvalue

$$\lambda^s = -\frac{a_0^2}{4} - \frac{\epsilon}{16\pi\rho_0} + O(\epsilon^2),$$

which using (8.39) yields the result of Corollary 8.6.1. Finally using the symmetry arguments, the principal eigenfunction for the Neumann-Dirichlet case is given by  $\phi(\frac{x}{2} + \pi) = \cos(\frac{x}{4})$  and using (8.48), the optimal mistuning

profile is  $k_m^*(x) = C \sin(\frac{x}{2})$ . This too is consistent with the results obtained using perturbation methods (see Corollary 8.6.2).

In order to compute the optimal mistuning profile, the ODE (8.49) is first simplified to

$$\frac{\phi_x}{\lambda^s - \frac{\epsilon C}{2\rho_0}(\phi_x)^2} d\phi_x = \frac{\phi}{a_0^2 + \frac{\epsilon C}{2\rho_0}\phi^2} d\phi,$$

that on integration gives

$$\lambda^s - \frac{\epsilon C}{2\rho_0}\phi_x^2 = \frac{D}{a_0^2 + \frac{\epsilon C}{2\rho_0}\phi^2}, \quad (8.51)$$

where  $D$  is a constant of integration. Using (8.50), we get  $D = \lambda^s \left( a_0^2 + \frac{\epsilon C}{2\rho_0} y_0^2 \right)$  where  $y_0 := \phi(\pi)$ . As a result, (8.51) becomes

$$\left[ 1 + \frac{\epsilon C}{2\rho_0 a_0^2} \phi^2(x) \right] \left[ 1 - \frac{\epsilon C}{2\rho_0 \lambda^s} \left( \frac{d\phi}{dx} \right)^2 \right] = \left[ 1 + \frac{\epsilon C}{2\rho_0 a_0^2} y_0^2 \right],$$

where  $y_0 = \phi(\pi)$ . After some manipulation, this equation yields the integral

$$\frac{a_0}{\sqrt{-\lambda^s}} \int_0^{\phi(x)} \left[ \frac{1 + \frac{\epsilon C}{2\rho_0 a_0^2} y^2}{y_0^2 - y^2} \right]^{\frac{1}{2}} dy = x. \quad (8.52)$$

The solution to this integral requires Elliptic functions of the second kind. In particular, we propose a co-ordinate change

$$\phi = y_0 \sin\left(\frac{\theta}{2}\right), \quad \theta \in [0, 2\pi]. \quad (8.53)$$

and define  $\beta := C y_0^2$ . Using (8.52),  $\theta$  is a solution to an implicit Elliptic integral equation

$$\int_0^\theta \left[ 1 + \frac{\epsilon \beta}{2\rho_0 a_0^2} \sin^2 \frac{\theta}{2} \right]^{\frac{1}{2}} d\theta = \frac{2\sqrt{-\lambda^s}}{a_0} x, \quad (8.54)$$

where substituting  $\theta = \pi$  and using (8.53), one obtains an implicit relationship between  $\beta$  and  $\lambda^s$ :

$$\int_0^{2\pi} \left[ 1 + \frac{\epsilon \beta}{2\rho_0 a_0^2} \sin^2 \frac{\theta}{2} \right]^{\frac{1}{2}} d\theta = \frac{4\pi\sqrt{-\lambda^s}}{a_0}, \quad (8.55)$$

In the  $\theta$  co-ordinate,

$$\frac{d\phi}{dx} = \left[ \frac{y_0}{2} \cos\left(\frac{\theta}{2}\right) \right] \frac{d\theta}{dx} = \frac{y_0 \sqrt{-\lambda^s}}{a_0} \cos\left(\frac{\theta}{2}\right) \left[ 1 + \frac{\epsilon\beta}{2\rho_0 a_0^2} \sin^2 \frac{\theta}{2} \right]^{-\frac{1}{2}}, \quad (8.56)$$

where  $d\theta/dx$  is obtained from (8.54). Using (8.48), (8.53), (8.55) and (8.56), we get the optimal pattern in the  $\theta$  co-ordinate as

$$k_m^*(\theta) = -A(\beta) \sin(\theta) \left[ 1 + \frac{\epsilon\beta}{2\rho_0 a_0^2} \sin^2 \frac{\theta}{2} \right]^{-\frac{1}{2}}, \quad (8.57)$$

where

$$A(\beta) = \frac{\beta}{8\pi} \int_0^{2\pi} \left[ 1 + \frac{\epsilon\beta}{2\rho_0 a_0^2} \sin^2 \frac{\theta}{2} \right]^{\frac{1}{2}} d\theta.$$

Now use the constraint  $\int_0^{2\pi} k_m^2 dx = 1$  (which implies  $\int_0^{2\pi} k_m(\theta)^2 (dx/d\theta) d\theta = 1$ ) to deduce the unknown constant  $\beta$ :

$$2\pi \frac{A^2(\beta)}{\int_0^{2\pi} \left[ 1 + \frac{\epsilon\beta}{2\rho_0 a_0^2} \sin^2\left(\frac{\theta}{2}\right) \right] d\theta} \int_0^{2\pi} \frac{\sin^2(\theta)}{1 + \frac{\epsilon\beta}{2\rho_0 a_0^2} \sin^2\left(\frac{\theta}{2}\right)} d\theta = 1$$

The eigenvalue  $\lambda^s$  is then obtained from (8.55),  $\theta(x)$  is obtained from (8.54), and the optimal mistuning pattern in the  $x$ -coordinate,  $k_m^*(x)$ , is deduced after substituting  $\theta(x)$  in (8.57). Figure 8.8 depicts a few typical optimal mistuning patterns for different values of  $\epsilon$ . Consistent with the results obtained using the perturbation method, the optimal mistuning pattern is close to the sinusoidal pattern for small values of  $\epsilon$ . Finally using the eigenvalue formula (8.55) and doing a little reduction, one also sees that the least stable eigenvalue approaches 0 as  $O(\frac{1}{N})$  for  $\epsilon > 0$ . Numerical evaluations of this exact formula appear in Figure 8.9.

# Chapter 9

## Summary

The estimation and control problems examined in this dissertation share the common attribute of being defined over graphs. The other common features are that only relative measurements are available. The underlying theme of our investigations is the effect of interconnect structure in large-scale systems. An important lesson learned from the results described here is that interconnection topology dictates, to a large extent, the achievable performance. Moreover, the matrix-valued effective resistance – introduced in this dissertation – has proven to be useful in analyzing the scalability of both estimation and control algorithms.

# Index

- 1-hop subgraph, 58
- 2-hop subgraph, 79
- active communication, 61
- adjacency matrix, 63
- algebraic connectivity, 176
- angle measurements, 22
- asymptotic distance ratio, 143
- asynchronous algorithm, 59
- basis combined incidence matrix, 63
- basis incidence matrix, 40
- boundary, 41
- camera calibration, 23
- centralized control, 172
- circulation, 125
- civilized drawing, 145
- clock offset, 25
- clock skew, 25
- combined incidence matrix, 62
- communication failure, 59
- communication graph, 49
- correctness, 52
- decentralized control, 173
- degree matrix, 62
- dense drawing, 144
- dense graph, 144
- directed path, 69
- Dirichlet Laplacian, 41
- distributed algorithm, 49
- divergence, 124
- drawing function, 142
- edge-function, 105
- effective resistance, 107
- electrical analogy, 109
- embedding, 115
- energy dissipation of a flow, 106
- error ratio, 52
- estimator, 37
- flow, 39, 105
- formation control, 171
- formation error, 178
- fuzz, 148
- generalized current, 105

generalized electrical network, 105  
 generalized incidence matrix, 33  
 graph drawing, 142  
 graphical distance, 56  
 in-degree matrix, 62  
 incidence matrix, 33  
 incidence operator, 124  
 incident, 32  
 infinite graph, 100  
 initial communication graph, 62  
 intensity, 39  
 interconnection graph, 172  
 inverse-positivity, 81  
 Jacobi algorithm, 57  
 Kirchoff's current law, 39, 105  
 Kirchoff's loop law, 105  
 Laplacian disagreement, 173  
 lattice, 137, 145  
 limiting BLUE covariance, 101  
 localization, 22  
 M-matrix, 68  
 matrix of class K, 68  
 matrix of class Z, 68  
 maximum connected range, 142  
 maximum uncovered diameter, 143  
 measurement graph, 32  
 measurement neighborhood, 56  
 measurement network, 36, 101  
 minimum node distance, 142  
 mistuning, 219  
 natural drawing, 152  
 nested graphs, 109  
 node failure, 59  
 node-function, 105  
 non-negative matrix, 69  
 Ohm's law, 105  
 orientation, 43  
 parallel edges, 118  
 range measurements, 22  
 Rayleigh's Monotonicity Law, 115  
 reference nodes, 32  
 regular graph, 152  
 relative offset measurement, 26  
 relative skew measurement, 27  
 spacing error, 199  
 sparse graph, 145  
 spectral radius, 66  
 subgraph, 100  
 swarming in nature, 172  
 symmetric communication, 49, 51

termination criteria, 58

time synchronization, 25

triangular lattice, 147

undirected path, 36

weakly connected, 36

weakly connected to a set, 36

weighted degree, 72

## Bibliography

- [1] J. Elson, L. Girod and D. Estrin. Fine-grained network time synchronization using reference broadcasts. In *the Fifth Symposium on Operating Systems Design and Implementation (OSDI)*. 2002.
- [2] P. G. Doyle and J. L. Snell. Random walks and electric networks. Math. Assoc. of America, 1984.
- [3] S. Fan, A. F. Blumberg, M. S. Bruno, D. Kruger, *et al.*. The skill of an urban ocean forecast system. In M. L. Spaulding (editor), *9th International Conference on Estuarine and Coastal Modeling*. 2005.
- [4] T. Samad, J. Bay and D. Godbole. Network-centric systems for military operations in urban terrain: The role of UAVs. *Proceedings of the IEEE*, vol. 95: 92–107, 2007.
- [5] M. Cantoni, E. Weyer, L. Yuping, S. K. Ooi, *et al.*. Control of large-scale irrigation networks. *Proceedings of the IEEE*, vol. 95: 75–91, 2007.
- [6] Automated highway systems. Kluwer Academic/Plenum Publishers, 1997.
- [7] J. K. Hedrick, M. Tomizuka and P. Varaiya. Control issues in automated highway systems. *IEEE Control Systems Magazine*, vol. 14: 21 – 32, 1994.
- [8] D. Estrin, R. Govindan, J. Heidemann and S. Kumar. Next century challenges: scalable coordination in sensor networks. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 263–270. ACM Press, 1999.
- [9] J. M. Mendel. *Lessons in Estimation Theory for Signal Processing, Communications and Control*. Prentice Hall, 1995.

- [10] D. Estrin, D. Culler, K. Pister and G. Sukhatme. Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, vol. 1, no. 1: 59–69, 2002.
- [11] J. M. Kahn, R. H. Katz and K. S. J. Pister. Emerging challenges: Mobile networking for "smart dust". *Journal Of Communication and Networks*, vol. 2: 188–196, 2000.
- [12] S. Mueller, R. P. Tsang and D. Ghosal. Multipath routing in mobile ad hoc networks: Issues and challenges. In *Performance Tools and Applications to Networked Systems, LNCS*, vol. 2965, pp. 209–234. Springer Berlin/Heidelberg, 2004.
- [13] A. Arora, R. Ramnath, E. Ertin, P. Sinha, *et al.*. Exscal: Elements of an extreme scale wireless sensor network. In *11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, pp. 102–108. 2005.
- [14] A. K. Chandra, P. Raghavan, W. L. Ruzzo, R. Smolensky, *et al.*. The electrical resistance of a graph captures its commute and cover times. In *Proc. of the 21st Annual ACM Symposium on Theory of Computing*, pp. 574–586. 1989.
- [15] D. Niculescu and B. Nath. Error characteristics of ad hoc positioning systems (APS). In *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, pp. 20–30. 2004.
- [16] A. Savvides, W. Garber, , R. Moses, *et al.*. An analysis of error inducing parameters in multihop sensor node localization. *IEEE Transactions on Mobile Computing*, vol. 4, no. 6: 567–577, 2005.

- [17] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, *et al.*. The effects of ranging noise on multihop localization: an empirical study. In *Proceedings of the 4th international symposium on Information processing in sensor networks, (IPSN'05)*, pp. 73–80. IEEE Press, Piscataway, NJ, USA, 2005.
- [18] E. Wagner, D. Jacques, W. Blake and M. Pachter. Flight test results of close formation flight for fuel savings. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*. 2002. AIAA-2002-4490.
- [19] M. J. Vachon, R. J. Ray, K. R. Walsh and K. Ennix. F/A-18 performance benefits measured during the autonomous formation flight project. Tech. rep., NASA Dryden Flight Research Center, 2002. Presented at the AIAA Atmospheric Flight Mechanics Conference, Monterey, California, August 5-8, 2002, AIAA-2002-4491.
- [20] R. G. Bill and W. F. Herrnkind. Drag reduction by formation movement in spiny lobsters. *Science*, vol. 193, no. 4258: 1146–1148.
- [21] D. Hummel. Aerodynamic aspects of formation flight in birds. *Journal of Theoretical Biology*, vol. 104: 321–347, 1983.
- [22] P. B. S. Lissaman and C. A. Shollenberger. Formation flight of birds. *Science*, , no. 3934: 1003 – 1005.
- [23] T. Balch and R. C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, vol. 14: 926 – 939, 1998.
- [24] S. R. Martin, R. M. Morgan, S. M. Gunter and R. D. Bartos. The starlight space interferometer: optical design and performance modeling. In M. Shao

- (editor), *Interferometry in Space. Proceedings of the SPIE*, vol. 4852 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pp. 500–511. 2003.
- [25] S. Martinez, J. Cortes and F. Bullo. Motion coordination with distributed information. *IEEE Control Systems Magazine*, 2007. To appear.
  - [26] B. Bamieh, F. Paganini and M. Dahleh. Distributed control of spatially invariant systems. *IEEE Transaction on Automatic Control*, vol. 47, no. 2, 2002.
  - [27] B. Bamieh and P. Voulgaris. A convex characterization of distributed control problems in spatially invariant systems with communication constraints. *Systems and Control Letters*, vol. 54: 575–583, 2005.
  - [28] C. Langbort, R. S. Chandra and R. D’Andrea. Distributed control design for systems interconnected over an arbitrary graph. *IEEE Transactions on Automatic Control*, vol. 49: 1502 – 1519, 2004.
  - [29] R. Olfati-Saber, J. A. Fax and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, vol. 95, no. 1, 2007.
  - [30] W. Ren, R. W. Beard and E. M. Atkins. Information consensus in multivehicle cooperative control. *IEEE Control Systems Magazine*, vol. 27: 71–82, 2007.
  - [31] R. E. Chandler, R. Herman and E. W. Montroll. Traffic dynamics: Studies in car following. *Operations Research*, vol. 6, no. 2: 165–184, 1958.
  - [32] P. Seiler, A. Pant and J. K. Hedrick. Disturbance propagation in vehicle strings. *IEEE Transactions on Automatic Control*, vol. 49: 1835–1841, 2004.

- [33] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, *et al.*. Wireless sensor networks for habitat monitoring. In *ACM International Workshop on Wireless Sensor Networks and Applications (WSNA'02)*. 2002.
- [34] D. M. Doolin and N. Sitar. Wireless sensors for wildre monitoring. In *SPIE Symposium on Smart Structures and Materials/ NDE*. 2005.
- [35] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, *et al.*. Monitoring civil structures with a wireless sensor network. *IEEE Internet Computing*, vol. 10: 26– 34, 2006.
- [36] N. Shrivastava, R. M. U. Madhow and S. Suri. Target tracking with binary proximity sensors: fundamental limits, minimal descriptions, and algorithms. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pp. 251–264. 2006. ISBN 1-59593-343-3.
- [37] N. Patwari and A. O. Hero. Location estimation accuracy in wireless sensor networks. 2002.
- [38] S. Lanzisera, D. Lin and K. Pister. RF time of flight ranging for wireless sensor network localization. In *Workshop on Intelligent Solutions in Embedded Systems (WISES)*. 2006.
- [39] N. B. Priyantha, A. K. L. Miu, H. Balakrishnan and S. J. Teller. The cricket compass for context-aware mobile applications. In *Mobile Computing and Networking*, pp. 1–14. 2001.
- [40] A. Kalis and T. Antanakopoulous. Direction finding in IEEE802.11 wireless networks. *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 5: 940–948, 2002.

- [41] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11: 1330–1334, 2000.
- [42] M. Quinn, R. Mudumbai, U. Madhow and B. S. Manjunath. Scalable algorithms for tracking using a network of cameras. 2007. Submitted for publication.
- [43] J. Bruck, J. Gao and A. Jiang. Localization and routing in sensor networks by local angle information. In *MobiHoc*. 2005.
- [44] L. Doherty, K. S. J. Pister and L. E. Ghaoui. Convex position estimation in wireless sensor network. In *IEEE INFOCOM*, pp. 1655–1663. 2001.
- [45] D. K. Goldenberg, P. Bihler, Y. R. Yang, M. Cao, *et al.*. Localization in sparse networks using sweeps. In *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*, pp. 110–121. ACM Press, New York, NY, USA, 2006. ISBN 1-59593-286-0.
- [46] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Comput. Networks*, vol. 43, no. 4: 499–518, 2003.
- [47] D. Moore, J. Leonard, D. Rus and S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems*. 2004.
- [48] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AOA. In *IEEE INFOCOM*. 2003.
- [49] N. Patwari, A. O. H. III and M. Perkins. Relative location estimation in

- wireless sensor networks. *IEEE Transactions in Signal Processing*, vol. 51, no. 8: 2137–2148, 2003.
- [50] A. Savvides, C. Han and M. B. Strivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Mobile Computing and Networking, MobiCom 2001, Rome, Italy*, pp. 166–179. 2001.
  - [51] A. Basu, J. Gao, J. Mitchell and G. Sabhnani. Distributed localization by noisy distance and angle information. In *Proc. of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*. 2006.
  - [52] K. Chintalapudi, A. Dhariwal, R. Govindan and G. Sukhatme. Ad-hoc localization using ranging and sectoring. In *IEEE Infocomm*. 2004.
  - [53] T. He, P. A. Vicaire, T. Yan, L. Luo, *et al.*. Achieving real-time target tracking using wireless sensor networks. *ACM Transaction on Embedded Computing System*, 2007. To appear.
  - [54] R. Naik, S. Biswas and S. Datta. Distributed sleep-scheduling protocols for energy conservation in wireless networks. In *38th Hawaii International Conference on System Sciences*. 2005.
  - [55] R. Mudumbai, J. Hespanha, U. Madhow and G. Barriac. Scalable feedback control for distributed beamforming in sensor networks. In *Int. Symp. on Information Theory (ISIT)*, pp. 137 –141. 2005.
  - [56] M. Maróti, B. Kusy, G. Simon and Á. Lédeczi. The flooding time synchronization protocol. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 2004.
  - [57] F. Chenavier and J. L. Crowley. Position estimation for a mobile robot using

- vision and odometry. In *IEEE Conference on Robotics and Automation*. 1992.
- [58] S. Shoval and J. Borenstein. Measuring the relative position and orientation between two mobile robots with binaural sonar. In *Presented at the ANS 9th International Topical Meeting on Robotics and Remote Systems*. 2001.
- [59] J. Borenstein, H. R. Everett and L. Feng. Where am i? sensors and methods for mobile robot positioning. 1996. Contributing authors: S. W. Lee and R. H. Byrne.
- [60] C. Godsil and G. Royle. *Algebraic Graph Theory*. Graduate Texts in Mathematics. Springer, 2001.
- [61] W.-K. Chen. *Applied Graph Theory*. North Holland Publishing Company, 1971.
- [62] D. G. Luenberger. *Optimization by Vector Space Methods*. John Wiley and Sons, 1969. ISBN 0471-18117X.
- [63] G. Wang, D. Turgut, L. Bölöni, Y. Ji, *et al.*. A simulation study of a MAC layer protocol for wireless networks with asymmetric links. In *International Conference On Communications And Mobile Computing*, pp. 929 – 936. 2006.
- [64] P. Barooah, J. Hespanha and A. Swami. On the effect of asymmetric communication on distributed time synchronization, 2007. Submitted to conference publication.
- [65] P. Barooah, N. M. da Silva and J. P. Hespanha. Distributed optimal estimation from relative measurements for localization and time synchronization.

In P. B. Gibbons, T. Abdelzaher, J. Aspnes and R. Rao (editors), *Distributed Computing in Sensor Systems DCOSS*, vol. 4026 of *LNCS*, pp. 266 – 281. Springer, 2006.

- [66] —. Distributed optimal estimation from relative measurements in sensor networks: Applications to localization and time synchronization. Tech. rep., University of California, Santa Barbara, 2006. URL <http://www.ccec.ece.ucsb.edu/~pbarooah/publications/TR3.html>.
- [67] P. Barooah and J. P. Hespanha. Distributed optimal estimation from relative measurements. In *Proceedings of the 3rd International Conference on Intelligent Sensing and Information Processing (ICISIP)*, pp. 226–231. 2005.
- [68] —. Estimation from relative measurements : Algorithms and scaling laws. *IEEE Control Systems Magazine*, 2007. In press.
- [69] V. Delouille, R. Neelamani and R. Baraniuk. Robust distributed estimation in sensor networks using the embedded polygon algorithms. In *3rd International Workshop on Information Processing in Sensor Networks*. 2004.
- [70] D. L. Mills. Internet time synchronization: the network time protocol. *IEEE transactions in communications*, vol. 39: 1482–1493, 1991.
- [71] S. Ganeriwal, R. Kumar and M. B. Srivastava. Timing-sync protocol for sensor networks. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 2003.
- [72] B. M. Sadler and A. Swami. synchronization in sensor networks: an overview. In *IEEE MILCOM*. 2006.
- [73] R. Karp, J. Elson, D. Estrin and S. Shenker. Optimal and global time

synchronization in sensornets. Tech. rep., Center for Embedded Networked Sensing, Univ. of California, Los Angeles, 2003.

- [74] A. Giridhar and P. R. . Kumar. Distributed time synchronization in wireless networks: Algorithms and analysis (I). In *45th IEEE Conference on Decision and Control*. 2006.
- [75] G. H. Golub and C. F. van Loan. *Matrix Computations*. The John Hopkins University Press, 3rd edn., 1996.
- [76] A. Frommer and H. Schwandt. A unified representation and theory of algebraic additive Schwarz and multisplitting methods. *SIAM Journal of Matrix Analysis and Applications*, vol. 18, no. 4: 893–912, 1997.
- [77] L. Eisner and V. Mehrmann. Convergence of block iterative methods for linear systems arising in the numerical solution of euler equations. *Numerische Mathematik*, vol. 59: 541–559, 1991.
- [78] A. Berman and R. J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences*. Computer Science and Applied Mathematics. Academic Press, 1979.
- [79] G. D. Poole. Generalized m-matrices and applications. *Mathematics of Computation*, vol. 29: 903–910, 1975.
- [80] M. Fiedler and V. Ptak. On matrices with non-positive off-diagonal elements and positive principal minors. *Czechoslovak Mathematical Journal*, vol. 12: 382 – 400, 1962.
- [81] F. Chung. Spectral graph theory. Regional Conference Series in Mathematics, Providence, R.I., 1997.

- [82] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, vol. 23: 298–305, 1973.
- [83] A. Frommer, H. Schwandt and D. B. Szyld. Asynchronous weighted additive Schwarz methods. *Electronic Transactions on Numerical Analysis*, vol. 5: 48–61, 1997.
- [84] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, vol. 38: 393–422, 2002.
- [85] B. Bougard, F. Catthoor, D. C. Daly, A. Chandrakasan, *et al.*. Energy efficiency of the IEEE 802.15.4 standard in dense wireless microsensor networks: Modeling and improvement perspectives. In *Proceedings of Design, Automation and Test in Europe (DATE)*, vol. 1, pp. 196–201. 2005.
- [86] M. M. Carvalho, C. B. Margi, K. Obraczka and J. Garcia-Luna-Aceves. Modeling energy consumption in single-hop IEEE 802.11 ad hoc networks. In *Proceedings of the 13th International Conference on Computer Communications and Networks (ICCCN)*, pp. 367 – 372. 2004.
- [87] R. Min, M. Bhardwaj, S. Cho, A. Sinha, *et al.*. Low-power wireless sensor networks. In *Keynote Paper, 28th European Solid-State Circuits Conference (ESSCIRC)*. Florence, Italy, 2002.
- [88] E. Shih, S. Cho, B. H. C. Fred S. Lee and A. Chandrakasan. Design considerations for energy-efficient radios in wireless microsensor networks. *Journal of VLSI Signal Processing*, vol. 37: 77–94, 2004.
- [89] W. Ye, J. Heidemann and D. Estrin. An energy-efficient mac protocol for wireless sensor networks. In *Proceedings of the IEEE Infocom*, pp. 1567–1576. 2002.

- [90] IEEE 802.15.4 specifications. URL <http://www.ieee802.org/>.
- [91] A. Ault, X. Zhong and E. J. Coyle. K-nearest-neighbor analysis of received signal strength distance estimation across environments. In *1st workshop on Wireless Network Measurements*. 2005.
- [92] A. Frommer and D. Szyld. On asynchronous iterations. *Journal of Comp. Appl. Math.*, 123, pp. 201–216, 2000.
- [93] L. Dieci and J. Lorenz. Block m-matrices and computation of invariant tori. *SIAM Journal on Scientific and Statistical Computing*, vol. 13, no. 4: 885–903, 1992.
- [94] L. Eisner and V. Mehrmann. Convergence of block iterative methods for linear systems arising in the numerical solution of euler equations. *Numerische Mathematik*, vol. 59: 541–559, 1991.
- [95] R. Nabben. On a class of matrices which arise in the numerical solution of euler equations. *Numerische Mathematik*, vol. 63: 411–431, 1992.
- [96] G. D. Poole. Generalized m-matrices and applications. *Mathematics of Computation*, vol. 29, no. 131: 903–910, 1975.
- [97] J. C. Strikwerda. A probabilistic analysis of asynchronous iteration. *Linear Algebra and its Applications*, vol. 349: 125–154, 2002.
- [98] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, vol. 50, no. 2: 169–182, 2005.
- [99] S. Resnick. *A Probability Path*. Birkhäuser Boston, 1st edn., 1999.

- [100] C. D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, 2000. ISBN 0898714540.
- [101] Citysense - an open, urban-scale sensor network testbed. 2007. URL [www.citysense.org](http://www.citysense.org).
- [102] A. Howard, L. E. Parker and G. S. Sukhatme. The SDR experience: Experiments with a large-scale heterogeneous mobile robot team. In *9th International Symposium on Experimental Robotics (ISER04)*. 2004.
- [103] H. Flanders. Infinite networks: I - resistive networks. *IEEE Transactions on circuit theory*, vol. CT - 18, no. 3: 326–331, 1971.
- [104] A. H. Zemanian. *Infinite Electrical Networks*. Cambridge Tracts in Mathematics. Cambridge University Press, Providence, R.I., 1991.
- [105] J. Cserti. Application of the lattice green’s function for calculating the resistance of and infinite network of resistors. *American Journal of Physics*, vol. 68, no. 10: 896–906, 2000.
- [106] D. Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, vol. 12: 81–95, 1993.
- [107] E. Wong. *Stochastic Processes in Information and Dynamical Systems*. Robert E. Krieger Publishing Company, 645 New York Avenue, Huntington, NY 11743, 1979. ISBN 0882759213.
- [108] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, vol. 45: 167–256, 2003.
- [109] M. Penrose. *Random Geometric Graphs*. Oxford Univ. Press, 2003.

- [110] R. Diestel. *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Heidelberg, 3rd edn., 2005. ISBN 3-540-26182-6.
- [111] P. Barooah and J. P. Hespanha. Estimation from relative measurements: Error bounds from electrical analogy. In *Proc. of the 2nd International Conference on Intelligent Sensing and Information Processing(ICISIP)*, pp. 88–93. 2005.
- [112] C. Chang and A. Sahai. Cram  r-rao-type bounds for localization. *EURASIP Journal on Applied Signal Processing*, pp. 1–13, 2006.
- [113] R. Solis, V. S. Borkar and P. R. Kumar. A new distributed time synchronization protocol for multihop wireless networks. In *Proc. of the 45th IEEE Conference on Decision and Control*. 2006.
- [114] G. Narasimhan and M. Smid. Approximating the stretch factor of euclidean graphs. *SIAM Journal on Computing*, vol. 30, no. 3: 978–989, 2000.
- [115] M. L. Huson and A. Sen. Broadcast scheduling algorithms for radio networks. In *Military Communications Conference (MILCOM’95)*, vol. 2, pp. 647 – 651. 1995.
- [116] P. G. Doyle. Application of Rayleigh’s short-cut method to Polya’s recurrence problem. online, 1998. URL <http://math.dartmouth.edu/~doyle/docs/thesis/thesis.pdf>.
- [117] E. A. Jonckheere and P. Lohsoonthorn. *Geometric Topology of Networks*. Unpublished.
- [118] D. Atkinson and F. van Steenwijk. Infinite resistive lattices. *American Journal of Physics*, vol. 67: 486–492., 1999.

- [119] J. Roe. *Lectures on Coarse Geometry*, vol. 31 of *University lecture series*. American Mathematical Society, 2003.
- [120] F. Y. Wu. Theory of resistor networks: the two-point resistance. *Journal of Physics A : Mathematics and General*, vol. 37: 6653–6673, 2004.
- [121] L. Li, D. Alderson, J. C. Doyle and W. Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, vol. 2, no. 4: 431–523, 2005.
- [122] E. A. Jonckheere, M. Lou, J. P. Hespanha and P. Barooah. Effective resistance of gromov-hyperbolic graphs: Applications to asymptotic sensor network problems. 2007. Submitted to conference publication.
- [123] G. Korniss, M. Hastings, K. Bassler, M. Berryman, *et al.*. Scaling in small-world resistor networks. *Physics Letters A*, vol. 350: 324–330, 2006.
- [124] R. S. Smith and F. Y. Hadaegh. Control of deep space formation flying spacecraft; relative sensing and switched information. *AIAA Journal of Guidance, Control and Dynamics*, vol. 28, no. 1: 106–114, 2005.
- [125] J. K. Parrish, S. V. Viscido and D. Grnbaum. Self-organized fish schools: An examination of emergent properties. *Biological Bulletin*, vol. 202, no. 3: 296–305, 2002.
- [126] swarming. URL <http://en.wikipedia.org/wiki/Swarm>.
- [127] C. J. Cutts and J. R. Speakman. Energy savings in formation flight of pink-footed geese. *Journal of Experimental Biology*, vol. 189: 251261, 1994.
- [128] A. J. Fax and R. M. Murray. Information flow and cooperative control

- of vehicle formations. *IEEE Transactions on Automatic Control*, vol. 49: 1465–1476, 2004.
- [129] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, vol. 49: 1520–1533, 2004.
- [130] E. Justh and P. Krishnaprasad. Equilibria and steering laws for planar formations. *Systems and Control Letters*, vol. 52: 25–38, 2004.
- [131] J. A. Marshall, M. E. Broucke and B. A. Francis. Formations of vehicles in cyclic pursuit. *IEEE Transactions on Automatic Control*, vol. 49, 2004.
- [132] R. Pongvithithum, S. M. Veres, S. B. Gabriel and E. Rogers. Universal adaptive control of satellite formation flying. *International Journal of Control*, vol. 78: 45 – 52, 2005.
- [133] G. Lafferriere, A. Williams, J. S. Caughman and J. J. P. Veerman. Decentralized control of vehicle formations. *Systems & Control Letters*, vol. 54: 899–910, 2005.
- [134] L. W. Hong Shi and T. Chu. Virtual leader approach to coordinated control of multiple mobile agents with asymmetric interactions. *Physica D: Nonlinear Phenomena*, vol. 213: 51–65, 2006.
- [135] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, *et al.*. Collective memory and spatial sorting in animal groups. *Journal of theoretical Biology*, vol. 218: 1–11, 2002.
- [136] A. Okubo. Dynamical aspects of animal grouping: swarms, schools, flocks, and herds. *Advances in Biophysics*, vol. 22: 1–94, 1986.

- [137] L. Nottestad and B. E. Axelsen. Herring schooling manoeuvres in response to killer whale attacks. *Canadian Journal of Zoology*, vol. 77: 1540–1546, 1999.
- [138] A. Jadbabaie, J. Lin and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, vol. 48, no. 6: 988–1001, 2003.
- [139] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press., Cambridge, 1993.
- [140] R. Grone, R. Merris and V. S. Sunder. The laplacian spectrum of a graph. *SIAM Journal of Matrix Analysis and Applications*, vol. 11, no. 2: 218–238, 1990.
- [141] R. Merris. Laplacian matrices of graphs: A survey. *Linear Algebra and its Applications*, vol. 197-198: 143–176, 1994.
- [142] H. Zhang and Y. Yang. Resistance distance and kirchhoff index in circulant graphs. *International Journal of Quantum Chemistry*, vol. 107: 330–339, 2007.
- [143] A. Ghosh, S. Boyd and A. Saberi. Minimizing effective resistance of a graph. *SIAM Review (to appear)*, 2008.
- [144] H.-S. Tan, R. Rajamani and W.-B. Zhang. Demonstration of an automated highway platoon system. In *American Control Conference*, vol. 3, pp. 1823 – 1827. 1998.
- [145] S. Darbha, J. K. Hedrick, C. C. Chien and P. Ioannou. A comparison of spacing and headway control laws for automatically controlled vehicles. *Vehicle System Dynamics*, vol. 23: 597–625, 1994.

- [146] S. S. Stankovic, M. J. Stanojevic and D. D. Siljak. Decentralized overlapping control of a platoon of vehicles. *IEEE Transactions on Control Systems Technology*, vol. 8: 816–832, 2000.
- [147] S. Sheikholeslam and C. A. Desoer. Longitudinal control of a platoon of vehicles with no communication of lead vehicle information: A system level study. *IEEE Transactions on Vehicular Technology*, , no. 4, 1993.
- [148] R. Fenton. Automatic vehicle guidance and control- a state of the art survey. *IEEE Transactions on Vehicular Technology*, vol. VT-19, no. 1, 1970.
- [149] R. J. Caudill and W. L. Gerard. Vehicle-follower longitudinal control for automated transit vehicles. *Journal of Dynamic systems, Measurements and Control*, pp. 241–248, 1977.
- [150] L. P. Hajdu, K. W. Gardiner, H. Tamura and G. L. Pressman. Design and control considerations for automated ground transportation systems. *Proceedings of the IEEE*, vol. 56: 493– 513, 1968.
- [151] S. E. Shladover. Longitudinal control of automotive vehicles in close-formation platoons. *Journal of Dynamic systems, Measurements and Control*, vol. 113: 302–310, 1978.
- [152] S. Darbha and J. K. Hedrick. String stability of interconnected systems. *IEEE Transactions on Automatic Control*, vol. 41, no. 3: 349–356, 1996.
- [153] Y. Zhang, E. B. Kosmatopoulos, P. A. Ioannou and C. C. Chien. Autonomous intelligent cruise control using front and back information for tight vehicle following maneuvers. *IEEE Transactions on Vehicular Technology*, vol. 48: 319–328, 1999.

- [154] X. Liu, S. S. Mahal, A. Goldsmith and J. K. Hedrick. Effects of communication delay on string stability in vehicle platoons. In *IEEE International Conference on Intelligent Transportation Systems (ITSC)*. 2001.
- [155] W. S. Levine and M. Athans. On the optimal error regulation of a string of moving vehicles. *IEEE Transactions on Automatic Control*, vol. AC-11, no. 3: 355–361, 1966.
- [156] S. M. Melzer and B. C. Kuo. Optimal regulation of systems described by a countably infinite number of objects. *Automatica*, vol. 7, no. 3: 359–366, 1971.
- [157] K. C. Chu. Decentralized control of high-speed vehicle strings. *Transportation Science*, vol. 8: 361–383, 1974.
- [158] M. R. Jovanović and B. Bamieh. On the ill-posedness of certain vehicular platoon control problems. *IEEE Transactions on Automatic Control*, vol. 50, no. 9: 1307 – 1321, 2005.
- [159] L. E. Peppard. String stability of relative-motion PID vehicle control systems. *IEEE Transactions on Automatic Control*, pp. 579–581, 1974.
- [160] P. Barooah and J. P. Hespanha. Error amplification and disturbance propagation in vehicle strings. In *Proceedings of the 44th IEEE conference on Decision and Control*. 2005.
- [161] S. K. Yadlapalli, S. Darbha and K. R. Rajagopal. Information flow and its relation to the stability of the motion of vehicles in a rigid formation. In *Proceedings of the 2005 American Control Conference*, pp. 1853–1858. 2005.
- [162] D. Helbing. Traffic and related self-driven many-particle systems. *Review of Modern Physics*, vol. 73: 1067–1141, 2001.

- [163] S. Darbha and K. R. Rajagopal. Aggregation of a class of interconnected, linear dynamical systems. *Systems and Control Letters*, vol. 43, no. 5: 387–401, 2001.
- [164] M. E. Khatir and E. J. Davison. Decentralized control of a large platoon of vehicles using non-identical controllers. In *Proceedings of the 2004 American Control Conference*, pp. 2769–2776. 2004.
- [165] O. O. Bendiksen. Localization phenomena in structural dynamics. *Chaos, Solitons, and Fractals*, vol. 11: 1621–1660, 2000.
- [166] A. J. Rivas-Guerra and M. P. Mignolet. Local/global effects of mistuning on the forced response of bladed disks. *Journal of Engineering for Gas Turbines and Power*, vol. 125: 1–11, 2003.
- [167] B. Shapiro. A symmetry approach to extension of flutter boundaries via mistuning. *Journal of Propulsion and Power*, vol. 14, no. 3: 354–366, 1998.
- [168] P. G. Mehta, G. Hagen and A. Banaszuk. Symmetry and symmetry-breaking in wave equation with feedback. *SIAM Journal of Applied Dynamical Systems (accepted for publication)*, 2006.
- [169] S. M. Melzer and B. C. Kuo. A closed-form solution for the optimal error regulation of a string of moving vehicles. *IEEE Transactions on Automatic Control*, vol. AC-16, no. 1: 50–52, 1971.
- [170] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang. *Spectral Methods in Fluid Dynamics*. Springer Series in Computational Physics. Springer-Verlag, New York, 1983.
- [171] L. C. Evans. *Partial Differential Equations*, vol. 19 of *Graduate Studies in Mathematics*. American Mathematical Society, 1998.

- [172] A. S. Lewis and M. M. Overton. Eigenvalue optimization. *Acta Numerica*, pp. 149–190, 1996.
- [173] J. B. Keller. The shape of the strongest column. *Archive for Rational Mechanics and Analysis*, vol. 5: 275–285, 1960.
- [174] E. Zeidler. *Nonlinear Functional Analysis and its Applications I: Fixed-Point Theorems*. Springer-Verlag, New York, 1986.